

(19)日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11)特許出願公表番号

特表2000-516733

(P2000-516733A)

(43)公表日 平成12年12月12日(2000. 12. 12)

(51)IntCl.⁷
G 0 9 C 1/00識別記号
6 2 0F I
G 0 9 C 1/00

テーマコード(参考)

6 2 0 A

審査請求 未請求 予備審査請求 有 (全 69 頁)

(21)出願番号 特願平10-511051
 (86)(22)出願日 平成9年8月19日(1997.8.19)
 (85)翻訳文提出日 平成11年2月19日(1999.2.19)
 (86)国際出願番号 PCT/US97/15826
 (87)国際公開番号 WO98/08323
 (87)国際公開日 平成10年2月26日(1998.2.26)
 (31)優先権主張番号 60/024, 133
 (32)優先日 平成8年8月19日(1996.8.19)
 (33)優先権主張国 米国 (US)
 (81)指定国 EP(AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, L U, MC, NL, PT, SE), AU, CA, CN, I L, J P

(71)出願人 エヌティーアールユー クリプトシステムズ、インコーポレーテッド
 アメリカ合衆国、ロード アイランド 02860、ボータケット、レスター ウェイ、3番地
 (72)発明者 ホフスタイン ジェフリー
 アメリカ合衆国、ロード アイランド 02860、ボータケット、レスター ウェイ、3番地
 (74)代理人 弁理士 山本 恵一

最終頁に続く

(54)【発明の名称】 公開鍵暗号システム方法および装置

(57)【要約】

この公開鍵暗号システム符号化技法は、2つの数を法とする多項式代数および整約に基づく混合システムを使用し、それに対して、復号技法は、妥当性が基本確率理論に依存する非混合システムを使用する。デジタル・メッセージを符号化し復号する方法は、環Rのイデアルpおよびqを選択するステップ(305)と、環Rの要素fおよびgを生成するステップ(325)と、 $F_{sub} q$ 、すなわち、 $f \pmod q$ の逆数を生成するステップと、 $F_{sub} p$ 、すなわち $f \pmod p$ の逆数を生成するステップ(340)と、gおよび $F_{sub} q$ を使用して導出することのできる積と、qを法として合同であるhを含む公開鍵を作成するステップ(350)と、fおよび $F_{sub} p$ を導出することのできる専用鍵を作成するステップと、公開鍵およびランダム要素を使用してメッセージを符号化することによって符号化メッセージを作成するステップと、専用鍵を使用して符号化メッセージを復号することによって復号メッセージを作成するステップとを含む。

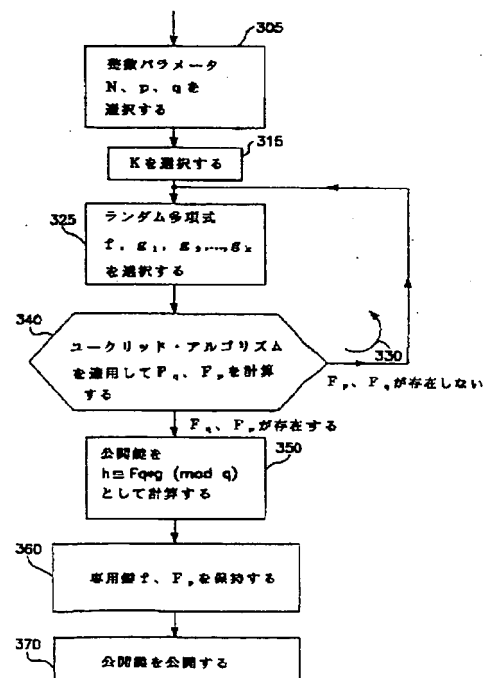


FIG. 3

【特許請求の範囲】

1. デジタル・メッセージ m を符号化し復号する方法であって、
 環 R のイデアル p および q を選択するステップと、
 環 R の要素 f および g を生成し、 $f \pmod{q}$ の逆数である要素 F_q を生成し、 $f \pmod{p}$ の逆数である要素 F_p を生成するステップと、
 g および F_q を使用して得ることのできる積と \pmod{q} で合同である h を含む公開鍵を生成するステップと、
 f および F_p を得ることのできる専用鍵を生成するステップと、
 専用鍵およびランダム要素 ϕ を使用してメッセージ m を符号化することによって符号化メッセージ e を生成するステップと、
 専用鍵を使用して符号化メッセージ e を復号することによって復号メッセージを生成するステップと
 を含む方法。
2. 前記環 R が環 Z を覆うモジュールであることを特徴とする請求の範囲第1項に記載の方法。
3. Z を覆う R の次元が N であり、 N が1より大きな整数であることを特徴とする請求の範囲第1項に記載の方法。
4. 環 R が、特定の多項式を法とする多項式の環であることを特徴とする請求の範囲第3項に記載の方法。
5. 要素を生成する前記ステップがさらに、 $g \pmod{q}$ の逆数である要素 G_q を生成し、 $g \pmod{p}$ の逆数である要素 G_p を生成するステップを含むことを特徴とする請求の範囲第1項に記載の方法。
6. 前記要素 G_q が前記公開鍵を得るのに使用され、前記要素 G_p が前記専用鍵の一部であることを特徴とする請求の範囲第5項に記載の方法。
7. 前記選択ステップがさらに正の整数 K を選択するステップを含み、前記要素 g が g_i ($i = 1, 2, \dots, K$) を含み、前記公開鍵 h が h_i ($i = 1, 2, \dots, K$) を含むことを特徴とする請求の範囲第1項に記載の方法。
8. 前記ランダム要素 ϕ がイデアル p 中に ϕ_i ($i = 1, 2, \dots, K$) を含み、前

記符号化メッセージが、

$$e \equiv \sum_{i=1}^k \phi_i * h_i + m \pmod{q}$$

として生成されることを特徴とする請求の範囲第7項に記載の方法。

9. 前記公開鍵および専用鍵がそれぞれ、 p と q をふくむことを特徴とする請求の範囲第1項に記載の方法。

10. 前記イデアル p と q が相対的に素な整数によって生成されることを特徴とする請求の範囲第1項に記載の方法。

11. 符号化メッセージが、メッセージ m と、 ϕ および h を含む積

との和と $\text{mod } q$ で合同であることを特徴とする請求の範囲第10項に記載の方法。

12. 前記整数 p と q が等しくなく、 p と q がともに1より大きいことを特徴とする請求の範囲第10項に記載の方法。

13. 前記符号化メッセージが、あるユーザによってある場所で生成され、前記ある場所から別の場所に伝送され、前記別の場所であるユーザによって復号されることを特徴とする請求の範囲第1項に記載の方法。

14. デジタル・メッセージ m を符号化し復号する方法であって、

整数 p および q を選択するステップと、

多項式 f および g を生成するステップと、

逆数 F_q および逆数 F_p を決定するステップであって、

$$F_q * f \equiv 1 \pmod{q}$$

$$F_p * f \equiv 1 \pmod{p}$$

であるステップと、

p 、 q 、 h を含む公開鍵を生成するステップであって、

$$h \equiv F_q * g \pmod{q}$$

であるステップと、

f および F_p を含む専用鍵を生成するステップと、

専用鍵およびランダム要素 ϕ を使用してメッセージ m を符号化することによっ

て符号化メッセージ e を生成するステップと、

専用鍵を使用して符号化メッセージ e を復号することによって復号メッセージを生成するステップと

を含む方法。

15. 前記符号化メッセージ e が、

$$e \equiv p \phi * h + m \pmod{q}$$

として生成されることを特徴とする請求の範囲第14項に記載の方法。

16. 前記復号メッセージが、

$$a \equiv f * e \pmod{q}$$

を計算し、ついで復号メッセージ m' を

$$m' \equiv F_p * a \pmod{p}$$

として計算することによって生成されることを特徴とする請求の範囲第15項に記載の方法。

17. 多項式 f および g を生成する前記ステップが、正の整数 K を選択し、 K 個の多項式 g を g_1, g_2, \dots, g_K として生成するステップを含み、前記公開鍵が h_1, h_2, \dots, h_K を含み、

上式で

$$h_i \equiv F_q * g_i \pmod{q}, \quad i = 1, 2, \dots, K$$

であることを特徴とする請求の範囲第14項に記載の方法。

18. 前記符号化メッセージ e が、

$$e \equiv p \phi_1 * h_1 + p \phi_2 * h_2 + \dots + p \phi_K * h_K + m \pmod{q}$$

として生成され、

上式で $\phi_1, \phi_2, \dots, \phi_K$ が K 個のランダム多項式であることを特徴とする請求の範囲第17項に記載の方法。

19. 前記符号化メッセージが、あるユーザによってある場所で生成され、前記ある場所から別の場所に伝送され、前記別の場所であるユーザによって復号されることを特徴とする請求の範囲第14項に記載の方法。

20. モニック多項式 $M(X)$ が選択され、多項式の乗算が、まず多項式の通常の乗算を行い、次いで結果を $M(X)$ で割り、剰余だけを保持することによって実施されることを特徴とする請求の範囲第14項に記載の方法。

21. 非ゼロの整数 N が選択され、多項式の乗算が、 N を法とする指数を整約することによって実施されることを特徴とする請求の範囲第14項に記載の方法。

22. 前記多項式 f 、 g 、 m 、 ϕ が有界係数をもつように制約されることを特徴とする請求の範囲第14項に記載の方法。

23. 前記整数 q が、前記整数 p と、前記多項式 f 、 g 、 m 、 ϕ の次数と、前記 f 、 g 、 m 、 ϕ の係数に対する前記制約とによって決まる量よりも小さく選ばれることを特徴とする請求の範囲第22項に記載の方法。

24. 前記整数 q が、前記整数 p と、前記多項式 f 、 g 、 m 、 ϕ の次数と、前記 f 、 g 、 m 、 ϕ の係数に対する前記制約とによって決まる量よりも大きく選ばれることを特徴とする請求の範囲第22項

に記載の方法。

25. デジタル・メッセージを符号化し復号する方法であって、

相対的に素な整数 p および q を選択するステップと、

非ゼロの整数 K を選択するステップと、

整数係数を有し、 $w_i \equiv 0 \pmod{p}$ ($i = 1, 2, \dots, K$)である行列の環から $K+2$ 個の行列 f 、 g 、 g_1 、 w_2, \dots, w_K を生成するステップと、

前記行列の環から逆行列 F_p 、 F_q 、 G_p 、 G_q を生成するステップであって、

$$f F_p \equiv I \pmod{p}$$

$$f F_q \equiv I \pmod{q}$$

$$g G_p \equiv I \pmod{p}$$

$$g G_q \equiv I \pmod{q}$$

上式で I が単位行列であるステップと、

公開鍵を K 個の行列 (h_1, h_2, \dots, h_K) のリストとして生成するステップであって、

$$h_i \equiv F_q w_i G_q \pmod{q}, \quad i = 1, 2, \dots, K;$$

であるステップと、

専用鍵を行列 (f, g, F_p, G_p) として生成するステップと、

専用鍵およびランダム整数 $\phi_1, \phi_2, \dots, \phi_K$ を使用してメッセージ m を符号化することによって符号化メッセージ e を

$$e \equiv \phi_1 h_1 + \phi_2 h_2 + \dots + \phi_K h_K + m \pmod{q};$$

として生成するステップと、

$$a \equiv f e g \pmod{q}$$

および

$$b \equiv a \pmod{p}$$

を計算し、ついで復号メッセージ m' を

$$m' = F_p b G_p \pmod{p}$$

として計算することによって復号メッセージ m' を生成するステップとを含む方法。

26. 前記符号化メッセージが、あるユーザによってある場所で生成され、前記ある場所から別の場所に伝送され、前記別の場所であるユーザによって復号されることを特徴とする請求の範囲第25項に記載の方法。

27. 前記行列 $w_1, w_2, \dots, w_K, f, g, m$ が有界係数をもつように制約され、整数 $\phi_1, \phi_2, \dots, \phi_K$ が有界であるように制約されることを特徴とする請求の範囲第25項に記載の方法。

28. 前記整数 q が、前記整数 p と、前記整数 K と、前記多項式 $w_1, w_2, \dots, w_K, f, g, m$ の次数と、前記多項式 $w_1, w_2, \dots, w_K, f, g, m$ の係数に対する前記制約と、整数 $\phi_1, \phi_2, \dots, \phi_K$ に対する前記制約とによって決まる量よりも小さく選ばれることを特徴とする請求の範囲第27項に記載の方法。

29. 前記整数 q が、前記整数 p と、前記整数 K と、前記多項式 $w_1, w_2, \dots, w_K, f, g, m$ の次数と、前記多項式 $w_1, w_2, \dots, w_K, f, g, m$ の係数に対する前記制約と、整数 $\phi_1, \phi_2, \dots, \phi_K$ に対する前記制約とによって決まる量よりも大きく選ばれることを

特徴とする請求の範囲第27項に記載の方法。

30. デジタル・メッセージ m を符号化し復号するシステムであって、

イデアル p および q を選択する手段と、

環 R の要素 f および g を生成し、 $f \pmod{q}$ の逆数である要素 F_q を生成し、 $f \pmod{p}$ の逆数である要素 F_p を生成する手段と、

g および F_q を使用して得ることのできる積と \pmod{q} で合同である h を含む公開鍵を生成する手段と、

f および F_p を得ることのできる専用鍵を生成する手段と、

専用鍵およびランダム要素 ϕ を使用してメッセージ m を符号化することによって符号化メッセージ e を生成する手段と、

専用鍵を使用して符号化メッセージ e を復号することによって復号メッセージを生成する手段と

を備えるシステム。

31. 前記符号化メッセージが、あるユーザによってある場所で生成され、前記ある場所から別の場所に伝送され、前記別の場所であるユーザによって復号されることを特徴とする請求の範囲第30項に記載のシステム。

32. 通信システムのユーザ間で情報を通信する方法であって、

リング R と、 R 中のイデアル P および Q と、イデアル Q を法とする環 R に対する代表剰余系 C_Q の集合と、イデアル P を法とする環 R に対する代表剰余系 C_P の集合とを生成するステップと、

R 中の少なくとも2つの専用鍵要素 f_1, \dots, f_n および第1のユーザのイデアル Q の関数として環 R 中の少なくとも1つの公開鍵要素 h_1, \dots, h_k を生成するステップと、

環 R と、イデアル Q と、イデアル P と、 R 中の要素 h_1, \dots, h_k との記述を第1のユーザから第2のユーザに伝送するステップと、

イデアル P および Q と、公開鍵要素 h_1, \dots, h_k と、 R 中の専用メッセージ要素 m と、第2のユーザの少なくとも1つの専用ランダム要素 ϕ_1, \dots, ϕ_1 との関数として環 R 中の要素 e を生成するステップと、

e, f_1, \dots, f_n の関数 F を評価する R 中の結果 A を計算し、代表剰余系 C_Q の集合中の A の代表剰余系 a を計算し、 a, f_1, \dots, f_n の関数 G を評価する結果 B を計算し、代表剰余系 C_P の集合中の B の代表剰余系 b を計算し、 b, f_1, \dots, f_n の関数 H を評価する代表剰余系 C_P の集合中の結果 c を計算することによって、第1のユーザがメッセージ要素 m を決定できるように、要素 e を第2のユーザから第1のユーザに伝送するステップとを含む方法。

33. メッセージ要素 m が、 m が C_P の要素であるという条件を満たすことを特徴とする請求の範囲第32項に記載の方法。

34. a, b, c, f_1, \dots, f_n の関数を計算することによって、第1のユーザがメッセージ要素 m を決定することを特徴とする請求の範囲第32項に記載の方法。

35. 公開鍵要素 h_1, \dots, h_k が、1と k の間の各 i について要素 f_1 がイデアル Q を法として R 中で積 $h_i f_{k+1}$ と合同であるという

条件を満たすことを特徴とする請求の範囲第32項に記載の方法。

36. 専用鍵要素 f_1, \dots, f_{k+1} が、要素 f_1, \dots, f_k がイデアル P 中にあるという条件を満たすことを特徴とする請求の範囲第32項に記載の方法。

37. 専用ランダム要素 ϕ_1, \dots, ϕ_1 がイデアル P 中にあることを特徴とする請求の範囲第32項に記載の方法。

38. 公開鍵要素 h_1, \dots, h_k と、専用ランダム要素 $\phi_1, \dots, \phi_{k+1}$ と、専用メッセージ要素 m との関数として生成される要素 e が、イデアル Q を法として $\phi_1 h_1 + \phi_2 h_2 + \dots + \phi_k h_k + \phi_{k+1} m$ と合同である C_Q の要素として生成されることを特徴とする請求の範囲第32項に記載の方法。

39. e, f_1, \dots, f_n の関数 F を評価する結果 A が積 $e f_{k+1}$ であることを特徴とする請求の範囲第32項に記載の方法。

40. a, f_1, \dots, f_n の関数 G を評価する結果 B が要素 a であることを特徴とする請求の範囲第32項に記載の方法。

41. a, f_1, \dots, f_n の関数 H を評価する代表剰余系の C_P 集合中の結果 c が、

$c f_{k+1}$ がイデアル P を法として b と合同であるという条件を満たすことを特徴とする請求の範囲第32項に記載の方法。

42. 結果 c が、代表剰余系 C_P の集合中のメッセージ m の代表剰余系に等しいことを特徴とする請求の範囲第32項に記載の方法。

43. 環 R が、次数 N のモニック多項式 $M(X)$ によって生成される R のイデアルを法として1つの変数 X 中の多項式の環であり、 R のイデアル Q が整数 q によって生成されるイデアルであり、 R のイデアル P が整数 p によって生成されるイデアルであり、代表剰余系 C_Q の集合が、次数がせいぜい $N-1$ で q を法とする代表剰余系の固定した集合中の係数を有する R 中の多項式の集合であり、代表剰余系 C_P の集合が、次数がせいぜい $N-1$ で p を法とする代表剰余系の固定した集合中の係数を有する R 中の多項式の集合であることを特徴とする請求の範囲第32項に記載の方法。

44. 専用鍵要素 f_1, \dots, f_n 、 R 中の専用メッセージ要素 m 、および専用ランダム要素 ϕ_1, \dots, ϕ_1 が、その係数に対する境界を含むという条件を満たすことを特徴とする請求の範囲第43項に記載の方法。

45. 環 R が非可換体であることを特徴とする請求の範囲第32項に記載の方法。

46. 要素 h_1, \dots, h_k が、1と K の間の各 i について要素 $f_{k+1} h_i f_{k+2}$ がイデアル Q を法として R 中で f_i と合同であるという条件に従って C_Q 中で生成されることを特徴とする請求の範囲第32項に記載の方法。

47. 専用鍵要素 f_1, \dots, f_k がイデアル P 中にあることを特徴とする請求の範囲第32項に記載の方法。

48. 専用ランダム要素 $\phi_1, \dots, \phi_{2k+1}$ が、要素 ϕ_1, \dots, ϕ_k がイデアル P 中にあるという条件を満たすことを特徴とする請求の範囲第32項に記載の方法。

49. 公開鍵要素 h_1, \dots, h_k と、専用ランダム要素 $\phi_1, \dots, \phi_{2k+1}$ と、専用メッセージ要素 m との関数として生成される要素 e が、イデアル Q を法として $\phi_1 h_1 \phi_{k+1} + \phi_2 h_2 \phi_{k+2} + \dots + \phi_k h_k \phi_{2k} + \phi_{2k+1} + m$ と合同である C_Q の要

素として生成されることを特徴とする請求の範囲第45項に記載の方法。

50. 環 R が整数係数をもつ行列の環であり、 R のイデアル Q が、固定整数 q で割り切れるすべての行列からなるイデアルであり、 R のイデアル P が、固定整数 p で割り切れるすべての行列からなるイデアルであり、代表剰余系 C_q の集合が、 q を法とする代表剰余系の固定した集合中の係数を有する R の要素の集合であり、代表剰余系 C_p の集合が、 p を法とする代表剰余系の固定した集合中の係数を有する R の要素の集合であることを特徴とする請求の範囲第32項に記載の方法。

51. 専用鍵要素 f_1, \dots, f_n 、専用メッセージ要素 m 、および専用ランダム要素 ϕ_1, \dots, ϕ_1 が、その係数に対する境界を含むという条件を満たすことを特徴とする請求の範囲第50項に記載の方法。

52. 専用ランダム要素 ϕ_1, \dots, ϕ_1 が、要素 ϕ_1, \dots, ϕ_1 が単位行列の定数倍であるという条件を満たすことを特徴とする請求の範囲第50項に記載の方法。

53. 環 R が群 G の群環であり、 R のイデアル Q が、整数 q によって生成されるイデアルであり、 R のイデアル P が、整数 p によって生成されるイデアルであり、代表剰余系 C_q の集合が、 q を法とする代表剰余系の固定した集合中の係数を有する R の要素の集合であり、代表剰余系 C_p の集合が、 p を法とする代表剰余系の固定した集合中の係数を有する R の要素の集合であることを特徴とする請求の範囲第32項に記載の方法。

54. 専用鍵要素 f_1, \dots, f_n 、専用メッセージ要素 m 、および専用ランダム要素 ϕ_1, \dots, ϕ_1 が、その係数に対する境界を含むという条件を満たすことを特徴とする請求の範囲第53項に記載の方法。

55. 環 R が、二面関係 $X^N=1$ 、 $Y^2=1$ 、 $XY=YX^{N-1}$ に従うことを条件として2つの変数 X および Y 中の多項式の非可換環であり、 R のイデアル Q が整数 q によって生成されるイデアルであり、 R のイデアル P が整数 p によって生成されるイデアルであり、代表剰余系 C_q の集合が、次数がせいぜい $N-1$ で q を法とする代表剰余系の集合から選ばれた係数を有する変数 X における R 中の多項式の集合であり、代表剰余系 C_p の集合が、次数がせいぜい $N-1$ で p を法とする

代表剰余系の固定した集合から選ばれた係数を有する変数 X における R 中の多項式の集合であることを特徴とする請求の範囲第32項に記載の方法。

56. 専用鍵要素 f_1, \dots, f_n 、専用メッセージ要素 m 、および専用ランダム要素 ϕ_1, \dots, ϕ_1 が、その一部が、条件 $Y\phi = \phi Y$ を満たす R のすべての要素 ϕ からなる R の可換部分環 R_0 中にあるという条件を含む諸条件を満たすことを特徴とする請求の範囲第55項に記載の方法。

【発明の詳細な説明】

公開鍵暗号システム方法および装置

関連出願

本出願は、1996年8月19日に出願された米国仮特許出願第60/024133号の優先権を主張するものであり、この仮特許出願は引用によって本明細書に組み込まれる。

発明の分野

本発明は、情報の符号化および復号に関し、詳細には、プロセッサ・システムによってデジタル・メッセージを暗号化し復号する公開鍵暗号システムに関する。

発明の背景

2つの当事者、たとえば、2つのコンピュータの間でデータを安全に交換するには暗号化が必要である。現在使用されている一般的な暗号化方法には、専用鍵暗号化と公開鍵暗号化の2つがある。専用鍵暗号化では、2つの当事者が符号化および復号に使用される鍵を秘密裏に交換する。専用鍵暗号システムの広く使用されている例にはDES、すなわちデータ暗号化標準がある。このようなシステムは非常に高速であり、かつ非常に安全であるが、2つの当事者が鍵を秘密裏に交換しなければならないという欠点を有する。

公開鍵暗号システムは、各当事者が復号プロセスのセキュリティを損なわずに暗号化プロセスを公開することのできるシステムである。この暗号化プロセスは一般に落とし戸関数と呼ばれている。公開鍵暗号システムは、一般に専用鍵暗号システムよりも低速であるが、クレジット・カード番号など少量のデータを送信するために使

用され、また、専用鍵暗号化に使用される専用鍵を送信するためにも使用される。

従来、公開鍵暗号システムには様々な落とし戸関数が提案され実施されている。

公開鍵暗号システムを作成するために使用されているある種類の落とし戸関数

では、ある群のべき乗が使用され、すなわち、ある群の要素が取り出され、群演算を使用してその要素のべき乗が反復的に求められる。最も頻繁に選択される群は、大きな素数 p および q の $p \cdot q$ を法とする乗法群である。ただし、楕円曲線、アーベル多様体、場合によっては非可換行列群など他の群も提案されている。しかし、この種の落とし戸関数は、それぞれ100桁程度の大きな素数を必要とし、鍵の作成が面倒であり、暗号化および復号に使用されるべき乗プロセスは多数の計算を必要とし、 N ビットからなるメッセージを暗号化し復号するには百桁の数の多数の乗算と N^3 個程度の演算を必要とする。

公開鍵暗号システムを作成するために使用されている第2の種類の落とし戸関数は、群中のどの数が平方であるか、すなわち通常は、大きな素数 p および q の $p \cdot q$ を法とする乗法群を判定するのが困難であることに基づく。第1の種類の場合と同様に、鍵作成が面倒であり、符号化および復号に多数の計算が必要であり、 N ビットからなるメッセージを符号化し復号するには N^3 個程度の演算が必要である。

第3の種類の落とし戸関数では、群中で離散対数問題が使用され、一般には、大きな素数 p を法とする乗法群または楕円曲線が使用される。この場合も、素数 p が少なくとも150桁を必要とし、 $p-1$ が大きな素因数を必要とするので、鍵作成は面倒である。このよ

うなシステムはべき乗を使用し、したがってこの場合も、 N ビットからなるメッセージを符号化し復号するには N^3 個程度の演算を必要とする。

公開鍵暗号システムを作成するために使用されている第4の種類の落とし戸関数は、ナップサック問題または部分集合和問題に基づく。このような関数は、部分群、通常は、加算される正の整数の部分群を使用する。この種の多くの公開鍵暗号システムは、格子整約技法を使用して破られており、したがって、もはや安全なシステムとはみなされていない。

公開鍵暗号システムを作成するために使用されている第5の種類の落とし戸関数は、誤り訂正符号、特にゴッパ符号に基づく。このような暗号システムは、有限体、一般には2つの要素を含む有限体上で線形代数を使用する。このような暗

号システムに対する線形代数アタックがあり、したがって、安全な暗号システムの鍵は400000ビット程度の大きな矩形行列である。これは大部分の応用分野で大きすぎる。

公開鍵暗号システムを作成するために使用されている第6の種類の落とし戸関数は、大きな次元Nの大きな格子で極めて短い基本ベクトルを見つけることが困難であることに基づく。このようなシステムの鍵は、 N^2 ビット程度の長さを有し、これは多くの応用分野では大きすぎる。また、このような格子整約公開鍵暗号システムは非常に新しく、したがって、そのセキュリティは完全には分析されていない。

したがって、大部分のユーザは、比較的短く容易に作成される鍵を比較的高速の暗号化プロセスおよび復号プロセスと組み合わせる公開鍵暗号システムを有することが望ましいと考えている。

本発明の目的は、鍵が比較的短く、かつ容易に作成され、符号化プロセスおよび復号プロセスを高速に実行することのできる公開鍵暗号化システムを提供することである。本発明の目的は、比較的低いメモリ要件を有し、かつセキュリティ・レベル、鍵長、符号化・復号速度、メモリ要件、帯域幅をかなり柔軟に兼ね合わせることを可能にする様々なパラメータに依存する公開鍵暗号化システムを提供することである。

発明の概要

本発明は、鍵長が他の一般的な公開鍵暗号システムの鍵長に匹敵する鍵を、ベクトルの大きな集合からほぼ無作為に選択することを可能にし、適切な（たとえば、現状では $=2^{80}$ ）セキュリティ・レベルを備え、最も一般に使用されている公開鍵暗号システム、すなわち前述のべき乗暗号システムよりも1桁ないし2桁程度高速の符号化プロセスおよび復号プロセスを提供する。

本発明の公開鍵暗号システムの実施形態の符号化技法は、2つの数pおよびqを法とする多項式代数および整約に基づく混合システムを使用し、それに対して復号技法は、妥当性が基本確率理論に依存する非混合システムを使用する。本発明の公開鍵暗号システムのセキュリティは、多項式混合システムとpおよびqを

法とする規約化の独立性との相互作用によってもたらされる。セキュリティは、実験によって観測されるように、たいていの格子では、最も短いベクトルよりもわずかに長いに過ぎない多数のベクトルがある場合に最も短いベクトルを見つけることが非常に困難であることにも依存する。

本発明の実施形態は、環 R のイデアル p および q を選択するステップと、環 R の要素 f および g を生成するステップと、要素 F_p 、

すなわち $f \pmod{q}$ の逆数を生成し、 F_p 、すなわち $f \pmod{p}$ の逆数を生成するステップと、 g および F_q を使用して得ることのできる積と $m \pmod{q}$ で合同である h を含む公開鍵を生成するステップと、 f および F_p を得ることのできる専用鍵を生成するステップと、公開鍵およびランダム要素 ϕ を使用してメッセージ m を符号化することによって符号化メッセージ e を生成するステップと、専用鍵を使用して符号化メッセージ e を復号することによって復号メッセージを生成するステップとを含む、デジタル・メッセージ m を符号化し復号する方法の形態である。

本発明の他の特徴および利点は、以下の詳細な説明と添付の図面から容易に明らかになる。

図面の簡単な説明

図1は、本発明の実施形態を実施することのできるシステムのブロック図である。

図2は、この流れ図で引用された補助流れ図と共に、本発明の実施形態を実施することのできる公開鍵暗号化システムの流れ図である。

図3は、公開鍵および専用鍵を生成するための、本発明の実施形態によるルーチンの流れ図である。

図4は、公開鍵を使用してメッセージを符号化するための、本発明の実施形態による流れ図である。

図5は、専用鍵を使用して符号化メッセージを復号するための、本発明の実施形態による流れ図である。

図6は、公開鍵および専用鍵を生成する、本発明の他の実施形態によるルーチ

ンの流れ図である。

図7は、公開鍵を使用してメッセージを符号化するための、本発

明の他の実施形態による流れ図である。

図8は、専用鍵を使用して符号化メッセージを復号するための、本発明の他の実施形態による流れ図である。

詳細な説明

図1は、本発明の実施形態を実施する際に使用できるシステムのブロック図である。2つのプロセッサ・ベース・サブシステム105および155は、セキュリティの保障されないチャネル50、たとえば電話やインターネット通信チャネルなどの有線通信チャネルまたは無線通信チャネルを介して通信するように示されている。サブシステム105はプロセッサ110を含み、サブシステム155はプロセッサ160を含む。プロセッサ110および160とそれに関連する回路を後述のようにプログラムすると、これらを使用して本発明の実施形態を実装し、本発明の方法の実施形態を実現することができる。プロセッサ110および160はそれぞれ、任意の適切なプロセッサ、たとえば電子デジタル・プロセッサまたはマイクロプロセッサでよい。任意の汎用プロセッサまたは特殊目的プロセッサ、あるいは本明細書に記載した機能を電子的または光学的に、あるいは他の手段によって実行することのできる他の機械または回路を使用することが理解されよう。プロセッサはたとえば、Intel Pentiumプロセッサでよい。サブシステム105は通常、メモリ123、クロック・タイミング回路121、入出力機能118、モニタ125を含み、これらはすべて従来型の種類のものでよい。入力には、103で示したキーボード入力を含めることができる。通信はトランシーバ135を介して行われ、トランシーバ135は、モデム、または信号を伝達する任意の適切な装置を備えることができる。

この例示的な実施形態のサブシステム155は、サブシステム105と同様な構成を有することができる。プロセッサ160は、関連する入出力回路164、メモリ168、クロック・タイミング回路173、モニタ176を有する。入力

にはキーボード155が含まれる。サブシステム155と外部との通信はトランシーバ162を介して行われ、この場合も、トランシーバ162はモデム、または信号を伝達する任意の適切な装置を備えることができる。

本発明の公開鍵暗号システム実施形態の符号化技法は、2つの数 p および q を法とする多項式代数および整約に基づく混合システムを使用し、それに対して復号技法は、妥当性が基本確率理論に依存する非混合システムを使用する。[多項式が順序係数の好都合な表現（いくつかの係数がゼロであってよい、 N 個の順序係数を有する $N-1$ 次の多項式）であり、プロセッサが、指定された演算を係数に対して実行することが理解されよう。] 本発明の公開鍵暗号システムのセキュリティは、多項式混合システムと p および q を法とする規約化の独立性との相互作用によってもたらされる。セキュリティは、実験によって観測されるように、たいていの格子では、最も短いベクトルよりもわずかに長いに過ぎない多数のベクトルがある場合に最も短いベクトルを見つけることが非常に困難であることにも依存する。

本発明の暗号システムは、M. Blum他著「An Efficient Probabilistic Public-Key Encryption Scheme Which Hides All Partial Information」(Advances in Cryptology: Proceedings of CRYPTO 84, Lecture Notes in Computer Science, 第196巻、Springer-Verlag、1985年、289ページないし299ページ) およびS. Goldwasser他著「Probabilistic Encryption」J. Computer and System

Science 28(1984年)、270ページないし299ページ)に記載された確率暗号システムの一般的な枠組みに適合する。このことは、暗号化がランダム要素を含み、したがって各メッセージが多数の可能な暗号化を有することを意味する。符号化および復号ならびに鍵作成は、本発明の技法を使用して比較的高速にかつ容易に行われ、長さ N のメッセージ・ブロックを符号化または復号するのに必要な演算は $O(N^2)$ 個であり、したがって、RSAで必要とされる $O(N^3)$ 個の演算よりもかなり高速である。鍵長は $O(N)$ であり、R. J. McEliece著「A Public-Key Cryptosystem Based On Algebraic Coding Theory」(JPL Pasadena, DS

N Progress Reports 42-44(1978年)、114ページないし116ページおよびO. Goldreich等著「Public-Key Cryptosystems From Lattice Reduction Problems」(MIT-Laboratory for Computer Science再版、1996年11月)に記載されたような他の「高速」公開鍵システムで必要とされる $O(N^2)$ 鍵長に匹敵する。

本発明の暗号システムの実施形態は、4つの整数パラメータ(N 、 K 、 p 、 q)と、整数係数を有する $N-1$ 次多項式の3つの集合 L_a 、 L_ϕ 、 L_m とに依存する。この実施形態は、環 $R = \mathbb{Z}[X] / (X^N - 1)$ で働く。要素 $F \in R$ は多項式またはベクトルとして書かれる。

$$F = \sum_{i=1}^N F_i X^{N-i} = [F_1, F_2, \dots, F_N]$$

星印「*」は R の乗算を示す。この星印乗算は巡回畳み込み積、すなわち次式の $F * G = H$ として明示的に与えられる。

$$H_k = \sum_{i=1}^{k-1} F_i G_{k-i} + \sum_{j=1}^N F_i G_{N+k-i} = \sum_{i+j=k \pmod{N}} F_i G_j$$

(たとえば) q を法とする乗算を実行する際、係数は q を法として整約される。詳細は、付録1を参照されたい。

以下に公開鍵暗号システムの発明による実施形態の例を示す。例示を容易にするために非常に小さな数を使用されており、したがって、この例は暗号に関して安全ではない。例と共に、二重括弧([[]])内のマテリアルとして、現在の条件の下で暗号に関して安全な実質的な暗号システムを形成する動作パラメータについて説明する。特定のセキュリティ・レベルを達成する動作パラメータの詳細な議論は付録1に記載されている。付録1には、本発明の暗号システムの実施形態の、様々なアタック・タイプに対する耐性についても記載されている。

本発明の実施形態で使用される対象は $N-1$ 次の多項式である。

$$a_1 X^{N-1} + a_2 X^{N-2} + \dots + a_{N-1} X + a_N$$

上式で、係数 a_1, \dots, a_N は整数である。本発明の「星印」乗算では、 x^N が1で置き換えられ、 x^{N+1} が X で置き換えられ、 x^{N-2} が x^2 で置き換えられ、以下同様である。[多項式を N 組の数

$$[a_1, a_2, \dots, a_N]$$

で表すこともできる。このような場合、星印積を畳み込み積とも呼ぶ。Nの値が大きい場合、実行するステップが N^2 個ではなく $N \log N$ 個程度である高速フーリエ変換法を使用して畳み込み積をより高速に計算することができる。]たとえば、 $N=5$ とし、2つの例示的な多項式を用いた場合、星印乗算は以下の数式を与える。

$$\begin{aligned} & (x^4+2x^2-3x+2) * (2x^4+3x^3+5x-1) \\ &= 2x^8+3x^7+4x^6+5x^5-6x^4+16x^3-17x^2+13x-2 \\ &= 2x^3+3x^2+4x+5x-6x^4+16x^3-17x^2+13x-2 \\ &= -6x^4+18x^3-14x^2+17x+3 \end{aligned}$$

[[安全なシステムはたとえば、 $N=167$ または $N=263$ を使用することができる]] [この実施形態は、 x^N-1 のすべての倍数からなるイデアルを法とする整数係数を含む多項式の環を使用する。より一般的には、異なるイデアルを法とする多項式を使用することができる。さらに一般的には、他の何らかの環Rを使用することができる。環およびイデアルの詳細については、たとえば、I. N. Herstein著「Topics in Algebra」を参照することができる。

この実施形態の他の態様では、イデアル q などの整数を法として多項式の係数が整約される。このことは基本的に、各係数を q で除し、係数をその剰余で置き換えることを意味する。たとえば、 $q=128$ であり、ある係数が2377である場合、2377を128で除すると18であり、剰余が73であるので、この係数は73で置き換えられる。しかし、「対称剰余」を使用した方が容易である。このことは、剰余が0でないし $q/2$ である場合、係数はそのままであるが、 $q/2$ でないし q である場合は係数から q が減じられることを意味する。したがって、 $q=128$ に対称剰余を使用する場合、 $-55=73-128$ であるので2377は-55で置き換えられる。

この剰余プロセスが実行されていることを示すには、3重等号(\equiv)を符号「 $\text{mod } q$ 」と共に使用する。以下に、2つの多項式の星印乗算を5を法とする規約化と組み合わせる例を示す。答えには対称剰余が使用される。

$$(x^4+2x^2-3x+2)*(2x^4+3x^3+5x-1)=-6x^4+18x^3-14x^2+17x+3$$

$$\equiv -x^4-2x^3+x^2+2x-2 \pmod{5}$$

本発明の実施形態によって（かつ例示を容易にするために、上記で指摘した小さな数を用いて）公開鍵暗号システムを作成する際、

第1のステップは整数パラメータ N 、 K 、 p 、 q を選択することである。以下に例を示す。

$$N=5, K=1, p=3, q=128$$

[[安全なシステムはたとえば、 $N=167$ 、 $K=6$ 、 $p=3$ 、 $q=2^{16}=65536$ を使用することができる]] 好ましくは、 p と q は互いに素であり、すなわち、1よりも大きな共通因子を有さない。イデアル p とイデアル q を互いに素にすることが望ましいことについては付録1に記載されている。以下のように、多項式のいくつかの集合が選択される。

$$L_g = \{\text{係数が } -2, -1, 0, 1, 2 \text{ である多項式}\}$$

$$L_\phi = \{2\text{つの } -1, 2\text{つの } 1, 1\text{つの } 0 \text{ を係数として有する多項式}\}$$

$$L_m = \{\text{係数が } -1, 0, 1 \text{ である多項式}\}$$

[[安全なシステムはたとえば、以下の集合を使用することができる。

$$L_g = \{\text{係数が } -177 \text{ ないし } 177 \text{ である多項式}\}$$

$$L_\phi = \{\text{係数が } 40 \text{ 個の } 1, 40 \text{ 個の } -1, \text{ 残りの } 0 \text{ である多項式}\}$$

$$L_m = \{\text{係数が } -3 \text{ ないし } 3 \text{ である多項式}\}$$

(注：多項式は $N-1$ 次を有し、したがって、この例の安全なパラメータの場合、多項式は166次を有する。さらに、符号化されている実際のメッセージ m の係数を p 、すなわちこの例では $p=3$ で除すると、このメッセージは剰余で構成される)]]。

集合 L_g は、暗号システムの鍵を作成するために使用され、集合 L_ϕ はメッセージを符号化するために使用され、集合 L_m は、可能なメッセージの集合である。たとえば、

$$2x^4-x^3+x-2 \text{ は集合 } L_g \text{ に存在し、}$$

$x^4-x^3-x^2+1$ は集合 L_ϕ に存在する。

鍵作成者D a nは、この例の鍵作成を実施するとき、集合 L_ϕ から2つの多項式 f および g を選択する。この簡略化された例では $K=1$ であり、したがって1つの多項式 g のみが存在する。D a nが次式を選択するものと仮定する。

$$f=x^4-x^3+2x^2-2x+1$$

$$g=x^4-x^3+x^2-2x+2$$

[[安全なシステムはたとえば、 $K+1$ 個の多項式 $f, g_1, \dots, g_K \in L_\phi$ を $K=6$ と共に使用することができる。]]

本発明の要件は、 f が、 q を法とする逆数と p を法とする逆数を有さなければならないことである。このことは、次式が成立するように多項式 F_q および F_p が存在しなければならないことを意味する。

$$F_q * f \equiv 1 \pmod{q} \text{ および } F_p * f \equiv 1 \pmod{p}$$

周知のユークリッド・アルゴリズムを使用して F_q および F_p を算出することができる。たとえば、本明細書の付録I Iを参照することができる。(いくつかの f が逆数を有さないことがある。この場合、D a nは最初に戻り、別の f を選択する。) 上記の例 f では、次式が成立する。

$$F_q = 103x^4 + 29x^3 + 116x^2 + 79x + 58$$

$$F_p = 2x^4 + 2x$$

これが f の正しい F_q であることを検査するには、以下の乗算を行うことができる。

$$\begin{aligned} F_q * f &= (103x^4 + 29x^3 + 116x^2 + 79x + 58) * (x^4 - x^3 + 2x^2 - 2x + 1) \\ &= 256x^4 + 256x - 127 \end{aligned}$$

$$\equiv 1 \pmod{128}$$

同様に、 F_p が正しいことを検査するには、以下の乗算を行うことができる。

$$\begin{aligned} F_p * f &= (2x^4 + 2x) * (x^4 - x^3 + 2x^2 - 2x + 1) \\ &= 6x^3 - 6x^2 + 6x - 2 \\ &\equiv 1 \pmod{3} \end{aligned}$$

これで、鍵作成者D a nが自分の公開鍵、すなわち、次式によって与えられる

多項式 h を作成する準備が完了した。

$$h \equiv F_a * g \pmod{q}$$

[[安全なシステムは、次式によって与えられる K 個の多項式 h_1, \dots, h_K を使用することができる。]]

$$h_i \equiv F_a * g_i \pmod{q}, i=1, 2, \dots, K, K=6$$

この例では続いて、 Dan が次式を計算する。

$$\begin{aligned} F_a * g &= (103x^4 + 29x^3 + 116x^2 + 79x + 58) * (x^4 - x^3 + x^2 - 2x + 2) \\ &= 243x^4 - 50x^3 + 58x^2 + 232x - 98 \\ &\equiv -13x^4 - 50x^3 + 58x^2 - 24x + 30 \pmod{128} \end{aligned}$$

Dan の公開鍵は以下の多項式である。

$$h = -13x^4 - 50x^3 + 58x^2 - 24x + 30$$

Dan の専用鍵は多項式対 (f, F_p) である。原則的に、 F_p は常に f から計算することができるので、多項式 f 自体が専用鍵として働くことができる。しかし、実際には、 Dan はおそらく、 F_p を事前に算出し保存しておく必要がある。

この例の次の部分における専用鍵による符号化について説明する。符号化側 $Cathy$ が Dan の公開鍵 h を使用して Dan にメッセージを送信する必要があると仮定する。 $Cathy$ は可能なメッセージの集合 L_m からあるメッセージを選ぶ。たとえば、 $Cath$

y が以下のメッセージを送信すると仮定する。

$$m = x^4 - x^3 + x^2 + 1$$

$Cathy$ は、このメッセージを符号化するとき、集合 L_ϕ から無作為に多項式 ϕ を選択する。たとえば、 $Cathy$ は次式を選択する。

$$\phi = -x^4 + x^3 - x^2 + 1$$

$Cathy$ は、無作為に選択したこの多項式 ϕ 、 Dan の公開鍵 h (ならびに p および q 、すなわち、公開鍵の一部)、 $Cathy$ の平文メッセージを使用し、以下の公式を使用して符号化メッセージ e を作成する。

$$e \equiv p \phi * h + m \pmod{q}$$

[[安全なシステムは K 個の公開鍵 h_1, \dots, h_K を、安全な例についての $K=6$ と

共に使用することができる。C a t h y は、メッセージを符号化するとき、集合 L_ϕ から K 個の多項式 ϕ_1, \dots, ϕ_K を無作為に選択し、次いで $e \equiv p\phi_1 * h_1 + p\phi_2 * h_2 + \dots + p\phi_K * h_K + m \pmod{q}$ を計算することによって符号化メッセージ e を作成することができる。]] 別法として、 h が $pF_a * g \pmod{q}$ と等しくされ、その場合、公式 $e \equiv \phi * h + m \pmod{q}$ を使用してメッセージを符号化することができる。この例では、C a t h y は以下の計算を行う。

$$\begin{aligned} p\phi * h + m &= 3(-x^4 + x^3 - x^2 + 1) * (-13x^4 - 50x^3 + 58x^2 - 24x + 30) \\ &\quad + (x^4 - x^3 + x^2 + 1) \\ &= -374x^4 + 50x^3 + 196x^2 - 357x + 487 \\ &\equiv 10x^4 + 50x^3 - 60x^2 + 27x - 25 \pmod{128} \end{aligned}$$

したがって、C a t h y の符号化メッセージは以下の多項式であり、

$$e = 10x^4 + 50x^3 - 60x^2 + 27x - 25$$

C a t h y はこの符号化メッセージを D a n に送信する。

この例の次の部分における専用鍵を使用した復号について説明する。D a n は、メッセージ e を復号するためにまず、専用鍵 f を使用して以下の多項式を計算する。

$$a \equiv f * e \pmod{q}$$

使用中の例では、D a n は以下の計算を行う。

$$\begin{aligned} f * e &= (x^4 - x^3 + 2x^2 - 2x + 1) * (10x^4 + 50x^3 - 60x^2 + 27x - 25) \\ &= -262x^4 + 259x^3 - 124x^2 - 13x + 142 \\ &\equiv -6x^4 + 3x^3 + 4x^2 - 13x + 14 \pmod{128} \end{aligned}$$

したがって、多項式 a は次式で表される。

$$a = -6x^4 + 3x^3 + 4x^2 - 13x + 14$$

次に、D a n は F_p 、すなわち自分の専用鍵の他方の半分を使用して以下の計算を行う。

$$F_p * a \pmod{p}$$

この結果が復号メッセージである。したがって、この例では、D a n は以下の計算を行う。

$$\begin{aligned}
F_p * a &= (2x^4 + 2x) * (-6x^4 + 3x^3 + 4x^2 - 13x + 14) \\
&= 34x^4 - 4x^3 - 20x^2 + 36x - 38 \\
&\equiv x^4 - x^3 + x^2 + 1 \pmod{3}
\end{aligned}$$

この復号がなぜ有効であるかの詳細な説明については、付録 I を参照することができる。

本発明の他の実施形態では、環は行列の環である。たとえば、環 $R =$ (整数係数を有する $M \times M$ 個の行列の環) を使用することができる。

R の要素を以下に示す。

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1M} \\ a_{21} & a_{22} & & a_{2M} \\ \vdots & & \ddots & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{MM} \end{pmatrix}$$

上式で係数 a_{ij} は整数である。加算および乗算は、行列に対して通常行われる加算および乗算であり、このプロセッサが行列のメンバーを、従来型の方式で記憶され処理される数として扱えることが理解されよう。 $N = M^2$ である場合、 R の行列は N 個の係数を有する。互いに素な整数 p と q が選択される。

この場合、 Dan は、専用鍵を作成するために、 R から $K + 2$ 個の行列を選択する。これらの行列を

$$f, \quad g, \quad w_1, \quad w_2, \dots, w_K$$

と呼ぶ。これらの行列は、 $f, \quad g, \quad w_1, \dots, w_K$ がかなり小さな係数を有し、あらゆる w_i が次式を満たすという特性を有するべきである。

$$w_i \equiv 0 \pmod{p}$$

(言い換えれば、あらゆる w_i のあらゆる係数が p の倍数である。) Dan は、自分の鍵を作成するときに、 p および q を法とする f および g の逆数を見つける必要がある。したがって、 Dan は、次式を満たす R 内の行列 F_p, F_q, G_p, G_q を見つける。

$$f F_p \equiv I \pmod{p}$$

$$f F_q \equiv I \pmod{q}$$

$$g G_p \equiv I \pmod{p}$$

$$g G_q \equiv I \pmod{q}$$

上式で、 I は $M \times M$ 識別行列である。一般に、これを実行するのは容易であり、何らかの理由で1つの逆数が存在しない場合でも、 Dan は新しい f または g を選択するだけでよい。

Dan の公開鍵は、以下の条件によって決定される K 個の行列 (h_1, h_2, \dots, h_K) の並びである。

$$h_i \equiv F_q w_i G_q \pmod{q} \quad (i = 1, 2, \dots, K)$$

(w_i が p を法とするゼロと合同であることに留意されたい。) Dan の専用鍵は4つの行列 (f, g, F_p, G_p) である。原則的に、 f および g のみを専用鍵として使用できるが、実際には F_p, G_p を事前に計算し記憶しておいた方が効率的である。

この行列の例の符号化について次に説明する。 $Cathy$ がメッセージ m を符号化する必要があるものと仮定する。メッセージ m は、 p を法とする係数を有する行列である。 $Cathy$ は、自分のメッセージを符号化するために、ある条件を満たすいくつかの整数 ϕ_1, \dots, ϕ_K を無作為に選択する。たとえば、これらの整数としては、和 $\phi_1 + \dots + \phi_K$ が所定の値 d と等しい非負整数を選択することができる。(ϕ_i が通常の数であり、行列ではないことに留意されたい。同様に、 ϕ_i は、識別行列の倍数とみなすことができ、したがって環 R のあらゆる要素と交換することができる。)

$Cathy$ は、自分の ϕ_i を選択した後、以下の規則によって符号化メッセージ e を作成する。

$$e \equiv \phi_1 h_1 + \phi_2 h_2 + \dots + \phi_K h_K + m \pmod{q}$$

次に、この行列の例の復号について説明する。 Dan が符号化メッセージ e を受信しており、それを解読する必要があるものと仮定する。 Dan はまず、次式を満たす行列 a を計算する。

$$a \equiv f e g \pmod{q}$$

Dan は、通常どおり、 $-q/2$ ないし $q/2$ (すなわち、ゼロ対称係数) や 0

ないし $q-1$ など、ある限られた範囲内で a の係数を選択する。

パラメータが適切に選択される場合、行列 a は以下の和に丁度等しくなる。

$$a = \phi_1 w_1 + \phi_2 w_2 + \dots + \phi_K w_K + fmg$$

(これは常に、 q を法として真であるが、重要な点は、 q が十分大きい場合に、上式が、 q を法とする場合だけではなく厳密な等式になることである。) Dan の次のステップは、たとえば次式のように、 p を法として a を整約することである。

$$b \equiv a \pmod{p}$$

w_i のすべての係数を p で除することができるので、これは、以下のことを意味する。

$$b \equiv fmg \pmod{p}$$

最後に、 Dan は以下の計算を行い、

$$F_p b G_p \pmod{p}$$

最初のメッセージ m を再生する。

前述の $M \times M$ 行列の実施形態は優れた動作時間を有する。符号化は、加算しか必要とせず、 M^2 個程度の演算を実行する。復号は $M \times M$ 行列の 2 回の行列乗算を必要とし、したがって、 M^3 個の演算を実行する。メッセージ長は M^2 程度であり、したがって、 N が固有メッセージ長 (すなわち、 $N = M^2$) を示す場合、行列実施形態は、符号化には $O(N)$ 個のステップを必要とし、復号には $O(N^{3/2})$ 個のステップを必要とする。これに対して、多項式実施形態は、符号化に $O(N^2)$ 個のステップを必要とし、復号に $O(N^2)$ 個のステップを必要とし、RSA 公開鍵システムは、符号化に $O(N^3)$ 個のステップを必要とし復号に $O(N^3)$ 個のステップを必要とする。

予備的な分析により、次元が $N^2 + N$ (またはそれ以上) である

格子を使用する必要があるのは、行列実施形態に対する固有格予アタックだけであることがわかっている。これは、多項式実施形態を解読するために使用される $2N$ 次元格子に対する著しいセキュリティの向上である。

暴力的（または潜在的なmeet-in-the-middle）アタックを回避するには、 ϕ_i のサンプル空間をかなり大きくし、たとえば 2^{100} ないし 2^{200} にする必要がある。しかし、こうするのは困難である。たとえば、 ϕ_i が、和 d を有する非負の値として選択された場合、サンプル空間は、

$$\binom{d+K-1}{K-1} = \frac{(d+K-1)!}{d!(K-1)!}$$

個の要素を有する。したがって、たとえば $K=15$ および $d=1024$ を選択した場合、 $2^{103.8}$ 個の要素を有するサンプル空間が得られる。

公開鍵サイズは $KM^2 \log_2(g)$ ビットであり、専用鍵サイズは $2M^2 \log_2(pq)$ ビットである。これらは共に実地的なサイズである。

図2は、公開鍵暗号化システムと共に使用できる基本手順を示し、本発明の実施形態による特徴を説明する参照される他の流れ図によって示されるルーチンを指す。ブロック210は、公開鍵情報および専用鍵情報の生成と、公開鍵の「公開」を表す。本発明の実施形態のルーチンについて図3の流れ図に関して説明する。この例では、この演算がプロセッサ・システム105で実行されるものと仮定することができる。公開鍵情報は、公開することができ、すなわち、公衆の任意のメンバー、または専用鍵保持者が暗号化メッセージを受信する必要がある所望の群に利用させることができる。通常、

必ずしも必要ではないが、公開鍵保持者およびその公開鍵のディレクトリが維持されている中央公開鍵ライブラリ施設またはウェブサイトで公開鍵を利用可能にすることができる。この例では、プロセッサ・システム155のユーザがプロセッサ・システム105のユーザに秘密のメッセージを送信する必要があり、プロセッサ・システム155のユーザがプロセッサ・システム150のユーザの公開済みの公開鍵を知っているものと仮定する。

ブロック220は、所期のメッセージ受信側の公開鍵を使用して平文メッセージを符号化するためにメッセージ送信側（すなわち、この例ではプロセッサ・システム155のユーザ）によって使用することのできるルーチンを表す。本発明の実施形態によるこのルーチンについては、図4の流れ図に関して説明する。こ

の場合、暗号化メッセージはチャンネル50（図1）を介して送信される。

図2のブロック260は、暗号化メッセージを復号して平文メッセージを再生するルーチンを表す。この例では、この機能は、専用鍵情報を使用するプロセッサ・システム105のユーザによって実行される。本発明の実施形態の復号ルーチンについては図5の流れ図に関して説明する。

次に、図3を参照すると、全体的に図2のブロック210で表された、公開鍵および専用鍵を生成するルーチンの流れ図が示されている。このルーチンはこの例では、プロセッサ・システム105のプロセッサ110をプログラムするために使用することができる。ブロック305は、整数パラメータ N 、 p 、 q の選択を表す。前述のように、 N は、生成される多項式 f および g_i の次数を決定し、 p および q はそれぞれ、星印積を生成する際に使用される2つのイデアルである。ブロック315は K 、すなわち使用される多項式 g

i の数を表す。上記の簡略化された例では、 K は1であり、例示的で比較的安全な特定のシステムが $K=6$ を使用できることに留意されたい。次に、ブロック325は無作為な多項式 f 、 g_1 、 $g_2 \dots g_K$ の選択を表す。係数はたとえば、乱数発生装置を使用して選択することができ、乱数発生装置は、利用可能なハードウェアまたはソフトウェアを使用して周知の方法で実装することができる。この実施形態では、各プロセッサ・システムが乱数発生装置を備え、乱数発生装置は図1ではそれぞれ、ブロック130および185で指定されている。

ブロック340は、選択済みの多項式 f の逆数 F_q および F_p が存在する場合に、それらの逆数を前述のように求めるためにユークリッド・アルゴリズムを応用することを表す。 F_p 、 F_q が存在しない場合、再びブロック325に入り、新しい多項式 f が選択される。ループ330は、定義済みの逆数を算出できる多項式が選択されるまで継続する。[所与の多項式について逆数が存在する確率は比較的高く、したがって一般に、この条件が満たされるまでにループ330を横断する回数は比較的少ないことが予期される。] 次いで、ブロック350に入る。このブロックは前述のように、

$$h = F_q * g \pmod{q}$$

による公開鍵 h の計算を表す。[$K > 1$ の場合、 $i = 1, 2, \dots, K$ について公開鍵構成要素 h_i がある。] ブロック360で表されるように、専用鍵は多項式 f 、 F_p として保持され、この場合、ブロック370で表されるように、公開鍵を公開することができる。

図4は、平文メッセージ m の符号化を実施するようにプロセッサ・システム155 (図1) のプロセッサ160などのプロセッサをプログラムするルーチンの、図2のブロック240によって一般

的に表される流れ図である。符号化されるメッセージが入力され(ブロック420)、ランダム多項式 ϕ が選択される(ブロック430)。[$K > 1$ の場合、 K 個の無作為多項式 $\phi_1, \phi_2, \dots, \phi_K$ が選択される。] この多項式は、前述のように集合 L の多項式でよく、ランダム係数は、任意のハードウェア手段またはソフトウェア手段、たとえば、乱数発生装置185によって選択することができる。次いで、符号化メッセージ e を次式のように算出することができる(ブロック450)。

$$e = p \phi * h + m \pmod{q}$$

上記で指摘したように、 K が1よりも大きい場合、符号化メッセージは $e \equiv p \phi_1 * h_1 + p \phi_2 * h_2 + \dots + p \phi_K * h_K + m \pmod{q}$ になる。この符号化メッセージは、チャネル50を介して鍵保持者、すなわち、この例ではプロセッサ・システム105のユーザに送信する(ブロック460)ことができる。

図5は、図2で全体的にブロック260によって表された、暗号化メッセージを復号する本発明の実施形態によるルーチンの流れ図である。ブロック530は暗号化メッセージ e の受信を表す。定義済みの多項式 f および F_p と整数 N 、 p 、 q とを含む保持されている専用鍵情報が取り出される(ブロック550)。次に、ブロック570は次式の計算を表している。

$$a \equiv f * e \pmod{q}$$

次いで、本明細書では m' として指定された復号メッセージを次式のように算出することができる(ブロック580)。

$$m' \equiv F_p * a \pmod{p}$$

図6、図7、図8は、前述の行列実施形態に関する流れ図である。図6は、全体的に図2のブロック210によって表された、公開鍵

および専用鍵を生成するルーチンの流れ図である。前述のように、このルーチンをこの例で使用して、プロセッサ・システム105のプロセッサ110をプログラムすることができる。ブロック605は、整数パラメータ N 、 p 、 q の選択を表し、 N は行列係数の数であり、 p および q は互いに素な整数である。ブロック615は K の選択を表し、 K を選択することによって行列の数が決定される。次に、ブロック625はランダム行列 f 、 g 、 w_1 、 w_2, \dots, w_K の選択と、 w_1 、 w_2, \dots, w_K がすべて、 p を法として0と合同であるという要件とを表す。この場合も、この目的のために乱数発生装置130（図1）を使用することができる。

ブロック640は、定義済みの行列 F_p 、 F_q 、 G_p 、 G_q の決定を表す。これらの行列が存在しない場合、再びブロック625に入り、新しい行列 f および g が選択される。ループ630は、定義済みの逆数を算出できる行列が選択されるまで継続する。次いで、ブロック650に入る。このブロックは、公開鍵、すなわち、以下の条件によって決定される K 個の行列 (h_1, h_2, \dots, h_K) のリストの算出を表す。

$$h_i \equiv F_q w_i G_q \pmod{q} \quad (i=1, 2, \dots, K)$$

ブロック660で表されているように、専用鍵は行列 (f, g, F_p, G_p) として保持され、その場合、ブロック670で表したように公開鍵を公開することができる。

図7は、この行列実施形態の技法を使用して平文メッセージ m の符号化を実施するようにプロセッサ・システム155（図1）のプロセッサ160などのプロセッサをプログラムする、全体的に図2のブロック240で表されたルーチンの流れ図である。符号化されるメッセージが入力され（ブロック720）、ランダム整数 ϕ_1 、

ϕ_2, \dots, ϕ_K が選択される（ブロック730）。これらの整数は乱数発生装置18

5 (図1) によって選択することができる。次いで、符号化メッセージ e を次式のように算出することができる(ブロック750)。

$$e \equiv \phi_1 h_1 + \phi_2 h_2 + \dots + \phi_k h_k + m \pmod{q}$$

この符号化メッセージは、チャネル50を介して鍵保持者、すなわち、この例ではプロセッサ・システム105のユーザに送信することができる(ブロック760)。

図8は、図2で全体的にブロック260によって表された、この行列実施形態によって暗号化メッセージを復号するルーチンの流れ図である。ブロック830は暗号化メッセージ e の受信を表す。定義済みの多項式 F 、 g 、 F_p 、 G_p と整数 N 、 p 、 q とを含む保持されている専用鍵情報が取り出される(ブロック850)。次に、ブロック870は次式の計算を表している。

$$a \equiv f e g \pmod{q}$$

次に、 a は、次式のように、 p を法として b に整約される(ブロック880)。

$$b \equiv a \pmod{p}$$

次いで、次式のように復号メッセージが算出される(ブロック890)。

$$m' \equiv F_p b G_p \pmod{p}$$

特定の好ましい実施形態を参照して本発明を説明したが、当業者には本発明の趣旨および範囲内の変形形態が企図されよう。たとえば、公開鍵または専用鍵を任意の適切な媒体、たとえば「スマート・カード」上に記憶し、符号化および／または復号を実行できるマイクロプロセッサをこのスマート・カードに備え、それによって

暗号化メッセージをスマート・カードへ伝達し、かつ／あるいはスマート・カードから伝達することができることが理解されよう。

NTRU：環ベースの公開鍵暗号システム

JEFFREY HOFFSTEIN、JILL PIPHER、JOSEPH H. SILVERMAN

要約 NTRU、すなわち新規の公開鍵暗号システムについて説明する。NTRUは、鍵がかなり短く容易に作成されること、高速であること、およびメモリ要

件が低いことを特徴とする。NTRUの符号化および復号は、基本確率理論に基づくクラスタ化原則と組み合わせられた、多項式代数によって提案されている混合システムを使用する。NTRU暗号システムのセキュリティは、多項式混合システムと、互いに素な2つの整数 p および q を法とする整約の独立性との相互作用によってもたらされる。

目次

- 0. はじめに
- 1. NTRUアルゴリズムの説明
- 2. パラメータの選択
- 3. セキュリティ分析
- 4. 実施にあたって考慮すべき点
- 5. NTRUの適度なセキュリティ・パラメータ
- 6. 他のPKCSとの比較
- 付録A. 基本補題

§ 0. はじめに

D i f f i eおよびH e l l m a n [4] が、どのようにすれば1方向関数を使用して効率的で計算コストの低い公開鍵暗号を作成で

きるかについて説明して以来、このようなシステムの作成にかなりの関心が払われている。現在の所、最も広く使用されている公開鍵システムはRSAである。RSAは、1978年にR i v e s t、S h a m i r、A d e l m a nによって作成されたものであり[10]、大きな数を因数分解することが困難であることに基づく。他のシステムには、誤り補正符号に依存するM c E l l i e c eシステム[9]と、格子整約問題の難点に基づくG o l d r e i c h、G o l d w a s s e r、H a l e v i [5]の最近のシステムが含まれる。

この論文では、新規の公開鍵暗号システムについて説明する。この暗号システムをNTRUシステムと呼ぶ。符号化手順では、多項式代数と2つの数 p および q を法とする整約に基づく混合システムを使用し、これに対して、復号手順では、妥当性が基本確率理論に依存する非混合システムを使用する。NTRU公開鍵

暗号システムのセキュリティは、多項式混合システムと、互いに素な2つの整数 p および q を法とする整約の独立性との相互作用によってもたらされる。セキュリティは、(実験によって観測されるように)、たいていの格子では、(適度に短いベクトルではなく) 極めて短いベクトルを見つけることが困難であることにも依存する。

この論文で発表する内容は、すでに広く配布されているが未公表の前刷り [7] とは2つの主要な点で異なる。第1に、より良好な動作特性を有するシステムを生成するために使用できる新しいパラメータ K を導入した。第2に、主として、eメールを介して Don Coppersmith、Johan Hastad、Adi Shamir から頂いた多数のコメントと最近の論文 [3] 中の多数のコメントに基づいて、格子ベースのアタックの分析を展開し明確

化した。この機会を利用して、この研究に関心を抱き援助していただいた上記の各氏に対して謝意を表したい。

NTRUは、[1] および [6] に記載されたように確率暗号システムの一般的な枠組みに適合する。このことは、暗号化がランダム要素を含み、したがって各メッセージが多数の可能な暗号化を有することを意味する。NTRUを用いた符号化および復号は極めて高速であり、鍵の作成は高速で容易である。詳細は第4章および第5章を参照されたいが、NTRUでは、長さ N のメッセージ・ブロックを符号化または復号するのに必要な演算が $O(N^2)$ 個であり、RSAで必要とされる $O(N^3)$ 個の演算よりもかなり高速であることに留意されたい。さらに、NTRU鍵長は $O(N)$ であり、[9, 5] など他の「高速」公開鍵で必要とされる $O(N^2)$ 鍵長に匹敵する。

§ 1 NTRUアルゴリズムの説明

§ 1. 1 表記法

NTRU暗号システムは、4つの整数パラメータ (N 、 K 、 p 、 q) と、整数係数を有する $N-1$ 次多項式の3つの集合 L_g 、 L_ϕ 、 L_m に依存する。環 $R = \mathbb{Z}[X] / (X^N - 1)$ を使用する。要素 $F \in R$ を多項式またはベクトルとして書く。

$$F = \sum_{i=1}^N F_i x^{N-i} = [F_1, F_2, \dots, F_N]$$

⊗を使用してRの乗算を示す。この星印乗算は、巡回畳み込み積と

して明示的に与えられる。

$$H_k = \sum_{i=1}^{k-1} F_i G_{k-i} + \sum_{j=k}^N F_i G_{N+k-j} = \sum_{i+j \equiv k \pmod{N}} F_i G_j \text{ の場合、 } F \otimes G = H$$

(たとえば) qを法とする乗算を行う際、qを法として係数を整約

する。

注 原則的に、積 $F \otimes G$ の計算には N^2 回の乗算が必要である。しかし、NTRUによる典型的な積の場合、FとGのうちの一方が小さな係数を有し、したがって、 $F \otimes G$ の計算は非常に高速である。一方、Nが大きいとみなされる場合は、高速フーリエ変換を使用して $O(N \log N)$ 個の演算で積 $F \otimes G$ を計算した方が高速である。

§ 1. 2 鍵の作成

NTRU鍵を作成する場合、Danは、 $K+1$ 個の多項式 $f, g_1, \dots, g_K \in L_q$ を無作為に選択する。多項式 f は、qおよびpを法とする逆数を有するという追加の要件を満たさなければならない。パラメータが適切に選択される場合、これはfの大部分の選択肢について真であり、これらの逆数の実際の計算は、修正されたユークリッド・アルゴリズムを使用して容易に行うことができる。これらの逆数を F_q および F_p によって示す。

$$F_q \otimes f \equiv 1 \pmod{q} \quad \text{および} \quad F_p \otimes f \equiv 1 \pmod{p} \quad (1)$$

Danは次に、以下の数量を計算する。

$$h_i \equiv F_q \otimes g_i \pmod{q} \quad 1 \leq i \leq K \quad (2)$$

Danの公開鍵は、以下に示す多項式の並びである。

$$(h_1, h_2, \dots, h_K)$$

D a nの専用鍵は単一の多項式 f である。ただし、実際には、D a nは F_p を記憶する必要がある。

§ 1. 3 符号化

C a t h y (符号化側) が D a n (復号側) にメッセージを送信する必要があるものと仮定する。C a t h y はまず、1組の平文 L_m からメッセージ m を選択する。次に、C a t h y は K 個の多項式 $\phi_1, \dots, \phi_K \in L_\phi$ を無作為に選択し、D a nの公開鍵 $(h_1, \dots, h$

$\kappa)$ を使用して以下の計算を行う。

$$e = \sum_{i=1}^K p \phi_i \otimes h_i + m \pmod{q}$$

これが、C a t h y が D a n に送信する符号化メッセージである。

§ 1. 4 復号

D a n が C a t h y からメッセージ e を受信しており、自分の専用鍵 f を使用してこのメッセージを復号するものと仮定する。これを効率的に行うために、D a n は第 1. 1 節で説明した多項式 F を事前に計算しておくべきである。

e を復号するために、D a n はまず以下の計算を行う。

$$a \equiv f \otimes e \pmod{q}$$

この場合、D a n は $-q/2$ ないし $q/2$ の間隔で a の係数を選択する。次に、D a n は、 a を整数係数を有する多項式とみなして、以下の計算を行うことによってメッセージを再生する。

$$F_p \otimes a \pmod{p}$$

注 パラメータ値が適切である場合、この復号手順によって最初のメッセージが再生される確率は極めて高い。しかし、ある種のパラメータ選択肢では、復号が失敗することがあり、したがっておそらく、各メッセージ・ブロックにいくつかの検査ビットを含めるべきである。復号が失敗する通常の原因は、メッセージのセンタリングが不適切であることである。この場合、D a n は、わずかに異なる間隔、たとえば x がある小さな (正または負の) 値である場合の $-$

$q/2 + x$ ないし $q/2 - x$ で $a \equiv f \otimes e \pmod{q}$ の係数を

選択することによってメッセージを回復することができる。xのどの値も機能しない場合には、間隔障害があり、メッセージを容易に復号することはできない。パラメータ値が適切に選択される場合に

は、間隔障害が起こることはまれであり、したがって、実際上間隔障害を無視することができる。

§ 1. 5 なぜ復号が機能するか

D a n が計算した多項式 a は以下の数式を満たす。

$$\begin{aligned} a = f \otimes e &= \sum_{i=1}^K f \otimes p\phi_i \otimes h_i + f \otimes m \pmod{q} \\ &= \sum_{i=1}^K f \otimes p\phi_i \otimes F_q \otimes g_i + f \otimes m \pmod{q} \quad (2) \text{より} \\ &= \sum_{i=1}^K p\phi_i \otimes g_i + f \otimes m \pmod{q} \quad (1) \text{より} \end{aligned}$$

この最後の多項式について検討する。

$$\sum_{i=1}^K p\phi_i \otimes g_i + f \otimes m$$

パラメータの選択が適切である場合は、(ほぼ常に) すべての係数が $-q/2$ と $q/2$ の間に位置し、したがって、 q を法として係数が整約される場合でもこの多項式は変化しない。このことは、D a

n が、 q を法とする $f \otimes e$ の係数を $-q/2$ ないし $q/2$ の間隔に整約するときに、以下の多項式を厳密に再生することを意味する。

$$\mathbb{Z}[X] / (X^N - 1) \text{ において } a = \sum_{i=1}^K p\phi_i \otimes g_i + f \otimes m$$

次いで、 p を法として a を整約すると、多項式 $f \otimes m \pmod{p}$ が与えられ、 F_p による乗算によってメッセージ $m \pmod{p}$ が取り込まれる。

§ 2 パラメータの選択

§ 2. 1 表記およびノルム推定

要素 $F \in R$ の幅を次式のように定義する。

$$\|F\|_{\infty} = \max_{1 \leq i \leq N} \{F_i\} - \min_{1 \leq i \leq N} \{F_i\}$$

我々の表記が示すように、これは、 R 上の一種の L^{∞} ノルムである。同様に、次式によって R 上の対称 L^2 を定義する。

$$\|F\|_2 = \left(\sum_{i=1}^N (F_i - \bar{F})^2 \right)^{1/2} \quad \text{ただし} \quad \bar{F} = \frac{1}{N} \sum_{i=1}^N F_i$$

(同様に、 $\|F\|_2 / \sqrt{N}$ は F の係数の標準偏差である。)

前提 任意の $\varepsilon > 0$ について、 ε 、 N 、 K に応じて定数 c_1 、 $c_2 > 0$ があり、したがって、無作為に選択された多項式 F_1, \dots, F_K 、 $G_1, \dots, G_K \in R$ について、これらの多項式が次式を満たす確率は $1 - \varepsilon$ よりも大きくなる。

$$c_1 \sum_{i=1}^K \|F_i\|_2 \cdot \|G_i\|_2 \leq \left\| \sum_{i=1}^K F_i \otimes G_i \right\|_{\infty} \leq c_2 \sum_{i=1}^K \|F_i\|_2 \cdot \|G_i\|_2 \quad (3)$$

もちろん、この前提は、比 c_2 / c_1 が小さな ε に対して非常に大きい場合には、実質的な観点から無用である。しかし、 N および K が適度に大きな値を有し、 ε の値が非常に小さい場合でも、定数 c_1 、 c_2 が極端な値を有することはないことが判明している。このことは、実験によって多数の状況で検証されており、この論文において理論的な証拠の概要を示す。

§ 2. 2 サンプル空間

典型的なサンプル空間の例として以下のものを使用する。

$$L_r = \{g \in R : g \text{ は } -(r-1) \text{ ないし } (r-1) / 2 \text{ の係数を有する}\}$$

$$L_{\phi} = \{\phi \in R : \phi \text{ は、} 1 \text{ に等しい } d \text{ 個の係数と、} -1 \text{ に等しい } d \text{ 個の係数を有し、残りは } 0 \text{ である}\}$$

$$L_s = \{m \in R : m \text{ は } -(s-1) \text{ ないし } (s-1) / 2 \text{ の係数を有する}\}$$

後で、セキュリティを達成するために r 、 d 、 s が満たさなければならない様々な制約があることが理解されよう。また、あらゆる ϕ

$\in L_\phi$ が L^2 ノルム $\|\phi\|_2 = \sqrt{2d}$ を有し、それに対して、平均要素 $g \in L_g$ および $m \in L_m$ が L^2 ノルム $\|g\|_2 = \sqrt{N(r^2-1)/12}$ および $\|m\|_2 = \sqrt{N(s^2-1)/12}$ を有することにも留意されたい。表記を容易に

するために、 L_g 、 L_ϕ 、 L_m の要素の平均 L^2 ノルムを L_g 、 L_ϕ 、 L_m と書くことにする。

厳密に必要なことではないが、 $L_m \approx p L_\phi$ という追加の仮定を行

う。この仮定によって、可能な格子アタックを分析すると共に、そのようなアタックを有効でなくすることが容易になる。一例として、

$d \approx N/4$ を選択すると仮定する。この場合、 $s \approx \sqrt{6} p$ が選択され

る。したがって、無作為に m の係数に p を加算し、 m の係数から p を減算することによって、 m に含まれる固有 $\text{mod } p$ 情報の「密度を高くする」必要がある。

§ 2. 3 復号基準

§ 1. 5 で説明したように、 $\|\sum p \phi_i \otimes g_i + f \otimes m\|_\infty < q$ であ

る場合、 $D a n$ は符号化メッセージ m を復号することができる。上記の前提の不等式 (3) を使用して (K の代わりに $K+1$ を用い、 ε として十分に小さな値を選択する) 以下の推定を行うことができる。

$$\begin{aligned} \left\| \sum_{i=1}^K p \phi_i \otimes g_i + f \otimes m \right\|_\infty &\leq c_2 \sum_{i=1}^K p \|\phi_i\|_2 \cdot \|g_i\|_2 + \|f\|_2 \cdot \|m\|_2 \\ &\approx c_2 L_g (K p L_\phi + L_m) \\ &\approx c_2 p L_g L_\phi (K+1) \quad (L_m \approx p L_\phi \text{ と仮定}) \end{aligned}$$

したがって、(確率 $1 - \varepsilon$ で) 復号を行うために、 $D a n$ は、以下の復号制約を満たすパラメータを選択する必要がある。

$$c_2 p L_g L_\phi (K+1) < q \quad (4)$$

§ 3 セキュリティ分析

§ 3. 1 Meet-in-the-middle アタック

説明を簡単にするために（かつアタッカを助けるために）、 $K =$

- 1 であり、したがって、符号化メッセージが $e \equiv \phi \otimes h + m \pmod{q}$ として表されるものと仮定する。Andrew Odlyzko は、 ϕ に対して使用することのできる meet-in-the-middle アタックがあると指摘しており、専用鍵 f にも同様なアタックが適用されることが考えられる。簡単に言えば、 f を $f = f_1 + f_2$ に分割し、 $f_1 \otimes e$ を $-f_2 \otimes e$ と突き合わせ、対応する係

数がほぼ同じ値を有するような (f_1, f_2) を見つける。したがって、（たとえば） 2^{80} のセキュリティ・レベルを得るには、約 2^{160} 個の要素を含む集合から f 、 g 、 ϕ を選択しなければならない。

§ 3. 2 多重送信アタック

やはり説明を簡単にするために、 $K = 1$ と仮定する。Cathy が、同じ公開鍵と異なるランダム ϕ を使用して単一のメッセージ m を数回にわたって送信する場合、アタッカ Betty は、メッセージの大部分を再生できると考えられる。

簡単に言えば、Cathy

が $e_i \equiv \phi_i \otimes h + m \pmod{q} \quad (i = 1, 2, \dots, r)$ を送信すると仮定した場合、Betty は $(e_i - e_1) \otimes h^{-1} \pmod{q}$ を計算し、それによって $\phi_i - \phi_1 \pmod{q}$ を再生することができる。

しかし、 ϕ の係数は非常に小さいので、Betty は $\phi_i - \phi_1$ のみを再生し、これから ϕ_1 の多数の係数を再生する。 r が適度なサイズ（たとえば、4 または 5）である場合でも、Betty は、brute force によってすべての可能性を試験

するのに十分な ϕ_1 を再生し、それによって m を再生する。したがって、基本メッセージの他の何らかのスクランブリングを行わずに多重送信を行うことは好ましくない。Betty が単一のメッセージをこのように復号する場合でも、この

情報は、他のメッセージを復号するうえで助けとならないことに留意されたい。

§ 3. 3 格子ベースの攻撃

まず格子整約に関するいくつかの語について説明する。格子整約の目標は、所与の格子M内の1つまたは複数の「小さな」ベクトルを見つけることである。理論的には、M内の最小のベクトルはしらみつぶしの探索によって見つけることができるが、実際には、Mの次元が大きい場合、これは不可能である。Schorr [11、12] およびその他によって様々な改良を施されたLenstra-Lenstra-Lovasz [8] のLLLアルゴリズムでは、多項式時間でMの最小ベクトルが見つかるが、次元の大きな（たとえば、 ≥ 100 ）たいていの格子では、最小ベクトルは見つからず、最小のLLL判定可能ベクトルと実際の最小ベクトルとの間のギャップは次元と共に指数関数的に増加するようである。格子攻撃に対するNTRUのセキュリティについて説明するために、大きな次元の格子に関する以下の3つの仮説を考える。

(H₁) たいていの格子Mでは、Mの最小非ゼロ・ベクトルの長さ $\sigma(M)$ は次式を満たす。

$$\sqrt{\frac{\dim(M)}{2\pi e}} \text{Disc}(M)^{1/\dim(M)} \leq \sigma(M) \leq \sqrt{\frac{\dim(M)}{\pi e}} \text{Disc}(M)^{1/\dim(M)}$$

したがって、 $v \in M$ が次式を満たす場合、

$$|v| \geq \sqrt{\frac{\dim(M)}{\pi e}} \text{Disc}(M)^{1/\dim(M)}$$

v はほぼ同じ長さの指数関数的に多数のベクトルに隠される。(H₂) 格子Mが、(H₁)によって説明した予期される最短ベクトルよりも小さなベクトル w を有するが、他の場合には「ランダムな」格子であるものと仮定する。 w が次式を満たす場合、

$$|w| \geq \kappa^{-\dim(M)} \sqrt{\frac{\dim(M)}{\pi e}} \text{Disc}(M)^{1/\dim(M)}$$

格子整約によって w が見つかる可能性は非常に低い。

(H₃) (H₂)の状況であると仮定する。この場合、格子整約方法によって算出される最小非ゼロ・ベクトル v_{LLL} はほぼ確実に次式を満たす。

$$|v_{LLL}| \geq K^{\dim(M)} |w|$$

注 仮説 (H₂) および (H₃) に現われる格子整約定数 k は、実験および経験によって決定しなければならない。これは、RSA PKCS を用いた場合と同様であり、セキュリティは、積 $p q$ を因数分解する現在の機能の推定に依存する。これは、[5] に記載された PKCS により類似しており、PKCS のセキュリティは、格子の小さな（ほぼ直交化された）基を見つけるのが困難なことに直接関連している。次元の大きな（ ≥ 100 ）格子を用いた実験によって、 $K = 1.5^{1/100}$ を使用できることがわかっている。（たとえば、[11] および [12] を参照されたい。）因数分解が進むにつれて、RSA PKCS で大きな素数を使用することが必要になり、したがって、格子の整約が進むにつれて、より小さな値の k とそれに対応するより大きなパラメータを NTRU で使用することが必要になるのは間違いない。また、700 よりも大きな次元の格子については仮説 (H₂) および (H₃) を仮定するだけでよい。このような高次元の格子では、場合によっては Schnorr のブ

ロック整約改良を含む LLL アルゴリズムでも長時間を必要とする。次元が約 300 の格子について仮説 (H₂) および (H₃) を仮定する場合、ずっと優れた動作特性を有する NTRU パラメータを選択することができる。

§ 3. 3. 1 鍵 f に対する小格子アタック

まず、恐らく最も自然な格子から始め、すなわち、任意の 1 つの h_i を選択し、 $h_i \otimes f \pmod{q}$ も小さいという特性を有する

小さなベクトル f を探索する。これを行うには、 $h_i = [h_{i1}, \dots, h_{iN}]$ とし、以下の行列の列によって生成される格子 M を検討する。

$$M = \left(\begin{array}{cccc|cccc} \lambda & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & \lambda & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda & 0 & 0 & \cdots & 0 \\ \hline h_{i1} & h_{i2} & \cdots & h_{iN} & q & 0 & \cdots & 0 \\ h_{i2} & h_{i3} & \cdots & h_{iN} & 0 & q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{iN} & h_{i1} & \cdots & h_{i,N-1} & 0 & 0 & \cdots & q \end{array} \right)$$

将来の表記に関する都合上、この行列を次式のように書く。

$$M = \begin{pmatrix} \lambda I & 0 \\ \tilde{h}_i & qI \end{pmatrix}$$

アタックを最適化するためにアタッカによって数量 λ が選択される。

M は次式を満たすと考えられる。

$$\dim(M) = 2N \text{ および } \text{Disc}(M) = \lambda^N q^N$$

考慮すべき2つの問題がある。第1の問題は、 M に短いベクトルとして埋め込まれる実際の鍵 f である。 M が以下の目標ベクトルを含み、

$$v_{\text{target}} = [\lambda f_N, \dots, \lambda f_1, g_{i1}, \dots, g_{iN}]$$

v_{target} がわかることによって f を再生できることに留意されたい。しかし、 v_{target} の長さを次式のように算出することができる。

$$|v_{\text{target}}|_2 = \sqrt{|\lambda f|_2^2 + |g_i|_2^2} = L_g \sqrt{\lambda^2 + 1}$$

仮説(H_1)によれば、 $|v_{\text{target}}|_2$ が以下の不等式を満たす場合、 f はアタックを受けても安全である。

$$|v_{\text{target}}|_2 \geq \sqrt{\frac{\dim(M)}{\pi e}} \text{Disc}(M)^{1/\dim(M)} = \sqrt{\frac{2N\lambda q}{\pi e}}$$

言い換えれば、次式が成立する必要がある。

$$L_g \sqrt{\lambda + \lambda^{-1}} \geq \sqrt{\frac{2Nq}{\pi e}}$$

アタッカは左辺を最小化したいので、アタッカの視点からの最適な λ は $\lambda = 1$ (

補題A. 1 参照) である。したがって、次式が成立する場合は安全である。

$$q \leq \frac{\pi \epsilon L_g^2}{N} \quad (5)$$

考慮すべき第2の点は、M内の他のある小さなベクトルによって、アタッカがメッセージを復号できるかどうかである。したがって、
任意の小さなベクトル $[f', g'] \in M$ は、 f' と $h_i \otimes f' \equiv$

$g' \pmod{q}$ が共に小さいという特性を有する。しかし、アタッカが以下の計算を行った場合、

$$e \otimes f' = \sum_{j=1}^K p \phi_j \otimes h_j \otimes f' + m \otimes f' \pmod{q}$$

q を法とする小さな係数を有するのは、 $j = i$ を含む項だけである。したがって、単一の h_i を小さくする f' は復号鍵としては働かない。このことは、すべての h_j を同時に検討していることを示し、

したがって次の格子に進む。

§ 3. 3. 2 鍵 f に対する大格子アタック

アタッカは、1つの h_i のみを使用するのではなく、 h_i のある部分集合を使用して格子を形成することができる。アタッカが、 h_1, \dots, h_k ($1 \leq k \leq K$) を使用し、以下の行列の列によって生成される格子Mを形成するものと仮定する。

$$\begin{pmatrix} \lambda I & 0 & 0 & 0 & \dots & 0 \\ \tilde{h}_1 & qI & 0 & 0 & \dots & 0 \\ \tilde{h}_2 & 0 & qI & 0 & \dots & 0 \\ \tilde{h}_3 & 0 & 0 & qI & \dots & 0 \\ \vdots & & & & \ddots & \vdots \\ \tilde{h}_k & 0 & 0 & 0 & \dots & qI \end{pmatrix}$$

(前の節の略号を使用している) この格子は次式を満たす。

$$\dim(M) = (k+1)N \text{ および } \text{Disc}(M) = \lambda^N q^{kN}$$

この格子は、(自明のショートハンドを使用する) 目標ベクトルを含む。

$$v_{\text{target}} = \{\lambda f, g_1, g_2, \dots, g_k\}$$

(より厳密に言えば、 f の座標を反転する必要がある。) この目標ベクトルは以下の長さを有する。

$$|v_{\text{arg}}|_2 = \sqrt{|\lambda f|_2^2 + |g_1|_2^2 + \cdots + |g_k|_2^2} = L_g \sqrt{\lambda^2 + k}$$

仮説 (H_2) によれば、 v_{arg} の長さが次式を満たしているかぎり、格子整約では v_{arg} を見つけることはできない。

$$\begin{aligned} |v_{\text{arg}}|_2 &\geq \kappa^{-\dim(M)} \sqrt{\frac{\dim(M)}{\pi e} \text{Disc}(M)^{1/\dim(M)}} \\ &= \kappa^{-(k+1)N} \sqrt{\frac{(k+1)N}{\pi e}} \cdot \lambda^{1/(k+1)} q^{k/(k+1)} \end{aligned}$$

したがって、次式が成立する場合にはアタックを受けても安全である。

$$L_g \sqrt{\lambda^{2k/(k+1)} + k\lambda^{-2/(k+1)}} \geq \kappa^{-(k+1)N} \sqrt{\frac{(k+1)N}{\pi e}} q^{k/(k+1)}$$

前述のように、アタッカは λ を選択して左辺を最小化する。この場合も、 $\lambda = 1$ の場合に最小値が得られ (補題 A. 1 参照)、したがって、次式が成立するかぎり仮説 (H_2) の下で実際の鍵は安全である。

$$q^{k/(k+1)} \leq \kappa^{(k+1)N} L_g \sqrt{\frac{\pi e}{N}} \quad (6)$$

§ 3. 3. 3 スプリアス鍵 f に対する大格子アタック

アタッカは、真の鍵 f を探索するのではなく、復号鍵として働く他の何らかの鍵 F を見つけることを試みることがある。 F 自体と各積 $h_j \otimes F \pmod{q}$ をスプリアス鍵にするには、これらを小

さくしなければならない。話を厳密にするために、アタッカが F を見つけ次式を計算するものと仮定する。

$$G_j \equiv h_j \otimes F \pmod{q} \quad (j = 1, 2, \dots, K)$$

次式の幅 (L^∞ ノルム) が一般に、

$$\phi_1 \otimes G_1 + \phi_2 \otimes G_2 + \dots + \phi_K \otimes G_K + m \otimes F$$

あるラッピング係数 W について少なくとも W_0 であることを知る必要がある (システムが安全であるには W をどのくらい大きくしなければならないかの問題については第4章で論じる)。

アタッカは、スプリアス鍵 F を見つけるために、第3.3.2節で説明した格子 M を取り出し、格子整約技法を使用して小さなベクトル v_{LLL} を見つける。 M 内の最小非ゼロ・ベクトルはベクトル $v_{\text{target}} = \{\lambda f, g_1, \dots, g_K\}$ であり、したがって、仮説 (H_3) によれば次式が成立する。

$$|v_{LLL}|_2 \geq k^{(K+1)N} |v_{\text{target}}|_2$$

$v_{LLL} = [\lambda F, G_1, G_2, \dots, G_K]$ とすると、次式が成立することがわかる。

$$\sqrt{\lambda^2 |F|_2^2 + |G_1|_2^2 + \dots + |G_K|_2^2} \geq k^{(K+1)N} L_g \sqrt{\lambda^2 + K}$$

格子整約によって得られるベクトル v_{LLL} は、サイズが多少とも無作為に分散する成分を有する。特に、すべての長さ $|\lambda F|_2, |G_1|_2, \dots, |G_K|_2$ がほぼ同じであり、したがって、(ほぼ) 次式が得られる。

$$|\lambda F|_2, |G_1|_2, \dots, |G_K|_2 \geq k^{(K+1)N} L_g$$

一方、これと (3) を使用して以下の推定を行うことができる。

$$\begin{aligned} & |\phi_1 \otimes G_1 + \phi_2 \otimes G_2 + \dots + \phi_K \otimes G_K + m \otimes F|_\infty \\ & \geq c_1 (|\phi_1|_2 \cdot |G_1|_2 + \dots + |\phi_K|_2 \cdot |G_K|_2 + |m|_2 \cdot |F|_2) \\ & = c_1 L_\phi (|G_1|_2 + \dots + |G_K|_2 + |F|_2) \\ & \geq c_1 (K+1) L_\phi L_g k^{(K+1)N} \end{aligned}$$

したがって、ラッピング係数 W を用いた場合、パラメータとして次式を満たすパラメータが選択されるかぎり、スプリアス鍵を得ることはできない。

$$W_0 \leq c_1 (K+1) L_\phi L_g k^{(K+1)N} \quad (7)$$

(これを復号不等式 (4) と比較することができる)

§ 3.3.4 個々のメッセージに対する大格子アタック

考慮しなければならない他の種類の格子アタックがある。アタッカは、あらゆるメッセージを復号する鍵を探索するのではなく、個々のメッセージを探索する格子を構築することができる。以下の格子について考える。この格子は第3.3.2節で使用した格子に類似している。以下の行列の列によって生成される格子をMとする。

$$\begin{pmatrix} \lambda I & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \lambda I & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \lambda I & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \lambda I & \cdots & 0 & 0 \\ \vdots & & & & \ddots & & \vdots \\ \vdots & & & & & \ddots & \vdots \\ p\tilde{h}_1 & p\tilde{h}_2 & p\tilde{h}_3 & \cdots & p\tilde{h}_k & qI \end{pmatrix}$$

この格子は次式を満たし、

$$\dim(M) = (K+1)N \text{ および } \text{Disc}(M) = \lambda^{KN} q^N$$

以下のベクトルを含む(自明の表記を使用する)。

$$[\lambda\phi_1, \lambda\phi_2, \dots, \lambda\phi_K, e-m]$$

この格子がこのベクトルを含むのは、符号化メッセージeが以下の規則に従って作成されたからである。

$$p\phi_1 \otimes h_1 + p\phi_2 \otimes h_2 + \dots + p\phi_K \otimes h_K + m \equiv c \pmod{q}$$

$e-m \pmod{q}$ の係数が小さくないので(8)が短いベクトルになる可能性が低いことは明らかである。しかし、アタッカはeの値を知っており、したがって、既知の非格子ベクトル $[0, 0, \dots, 0, e]$ に近いベクトルをMで探索することができる。探索中の格子ベクトルおよび既知の非格子ベクトルからの距離は、以下のベクトルの長さである。

$$v_{\text{targ}} = [\lambda\phi_1, \lambda\phi_2, \dots, \lambda\phi_K, -m]$$

これは、非斉次格子問題の例である。非斉次問題は斉次問題よりもいくらか難しくなる傾向があるが、重大な誤りを犯すのを回避するために、アタッカが非斉次問題を斉次問題を解くのとまったく同じ程度に解くことができると仮定する。したがって、アタッカが以下の長さのベクトルを見つけられるかどうかを調べる必

要がある。

$$|v_{\text{arg}}|_2 = L_\phi \sqrt{K\lambda^2 + p^2}$$

(あらゆる $m \in L_m$ およびあらゆる $\phi \in L_\phi$ について $|m|_2 = p$

$|\phi|_2$ であることを想起されたい) 仮説 (H₂) によれば、次式が成立する場合

$$|v_{\text{arg}}| \geq \kappa^{-\dim(M)} \sqrt{\frac{\dim(M)}{\pi e}} \text{Disc}(M)^{1/\dim(M)}$$

あるいは言い換えれば、次式が成立する場合には、アタックは失敗する。

$$L_\phi \sqrt{K\lambda^{2/(K+1)} + p^2 \lambda^{-2K/(K+1)}} \geq \kappa^{-(K+1)N} \sqrt{\frac{(K+1)N}{\pi e}} q^{1/(K+1)}$$

アタックは、 $\lambda = p$ (補題 A. 1 参照) を使用することによって左辺を最小化し、したがって、次式が成立する場合、アタックは失敗する。

$$q^{1/(K+1)} \leq \kappa^{(K+1)N} L_\phi \sqrt{\frac{\pi e}{N}} p^{1/(K+1)} \quad (9)$$

この数式は、これによって補われる (6) と比較することができる。

§ 3. 3. 5 格子アタック・パラメータ制約の要約

本節の前の部分では、様々な格子アタックについて説明し、これらのアタックが成功するのを妨げる、パラメータに対する制約を考案した。すべての制約を満たすパラメータの選択肢が存在するかどうかという問題が残っている。読者に好都合なように、本節のすべての不等式を、真の鍵 f の所有者がメッセージを復号する場合に必要な基本不等式 (4) と共にリストする。

$$c_2 p L_s L_\phi (K+1) < q \quad (4)$$

$$q \leq \frac{\pi e L_s^2}{N} \quad (5)$$

$$q^{k/(k+1)} \leq \kappa^{(k+1)N} L_s \sqrt{\frac{\pi e}{N}} \quad (1 \leq k \leq K \text{ の全てに対して}) \quad (6_k)$$

$$W_g \leq c_1 (K+1) L_\phi L_g K^{(K+1)N} \quad (7)$$

$$q^{1/(K+1)} \leq K^{(K+1)N} L_\phi \sqrt{\frac{\pi e}{N}} p^{1/(K+1)} \quad (9)$$

任意の固定値 c_1 、 c_2 、 p 、 $L_\phi > 0$ であり、 p 、 k 、 $W > 1$ である場合、これらの不等式には常に解 N 、 k 、 L_g 、 q が存在すると考えられる。次に、解を求める際に助けとなるいくつかの注意事項について述べる。

まず、これらの不等式を様々な方法で組み合わせる。まず、(4) と (7) を組み合わせると (ある代数計算の後で) 次式が与えられる。

$$(K+1)N \geq \frac{\log(c_1^{-1} c_2 p W)}{\log K} \quad (10)$$

(基本的に) c_1 、 c_2 、 k を自由に選択することはできず、 W が所望のセキュリティ・レベルに応じて 5 と 10 の間で選択されることに留意されたい。これによって、通常はかなり小さな p が選択される。この場合の重要な点は、(10) が $(K+1)N$ の下限を与え、これを超えるとほとんど制御できなくなることである。

次に、(4) と (5) を組み合わせると次式が得られる。

$$L_g > \frac{c_2 p (K+1) N}{\pi e} L_\phi \quad (11)$$

q を選択する際にいくらか融通を利かせるには、 L_g の値として、指定されたこの下限よりも (たとえば) 1.5 倍ないし 2 倍大きな値を選択するとよい。

たとえば、 L_ϕ および L_g が第 2.2 節で説明したような値であ

る場合、 $L_\phi = \sqrt{2d}$ であり、大部分の $g \in L_g$ は $\|g\|_2 \sim L_g =$

$\sqrt{N(r^2 - 1)/12}$ を満たす。したがって、(11) を使用して L_g を選択した後、 $r = \lceil L_g \sqrt{12/N} \rceil$ を選択することができ、この場合、大部

分の $g \in L_g$ は所望の L_g に非常に近い L^2 ノルムを有する。さらに、 L_g から要素を選択するのは符号作成者の Dan だけであり、このような選択は 1 度行うだけ

でよいので、 Dan が、ほぼ L_g のノルムを有する L_g で必要な $K+1$ 個の多項式を見つけるのは難しいことではない。長さの制約がある場合でも、実際には r^N が少なくとも 2^{500} になる傾向があるので、 L_g 内のそのような多項式の数は一タッカがしらみつぶしの探索を介して検査できるよりもはるかに多い。

§ 4 実施にあたって考慮すべき点

§ 4. 1 セキュリティ係数およびラッピング係数

アタッカが、格子整約によって生成されたスプリアス鍵を使用するときにはどの程度のラッピングを期待できるかを、ラッピング係数 W が制御することを想起されたい。 W が小さすぎ、たとえば $W=1.5$ である場合、アタッカは多数の（場合によっては最も多くの）係数を回復することができる。これは、これらの値が平均の周りに集中する傾向があるからである。厳密には、アタッカは N 個の未知の係数に関して（たとえば） $0.95N$ 個の一次方程式を再生し、この場合、暴力的探索によってアタックが終了する。

CoppersmithおよびShamir [3] は、 W がこれよりも大きなビットであり、たとえば $W=2.5$ である場合でも、アタッカはクラスタ化により、 N 個の未知の係数について約 $0.67N$ 個の一次方程式を得ることができると考えた。CoppersmithおよびShamirは次いで、アタッカが2つの独立のスプリアス鍵を構築し適用した場合、システムの解を求めるのに十分

な数の独立の等式を得ることができると考えている。CoppersmithおよびShamirはさらに、 $W=4$ である場合、いくつかの短いベクトルを使用し、ある種の誤り補正技法を使用することによって、アタックを成功させることができるが、 W が10程度である場合には、この種のアタックは成功しないと述べている。詳細については[3]を参照されたい。

これらのことに基づいて、ラッピング係数 $W=10$ を使用してサンプル動作パラメータを構築する。

§ 4. 2 サンプル動作パラメータ

この節では、第3節の仮説の下で安全であるNTRU PKCSの2組の使用可能なパラメータを考案する。これらのパラメータ集合によってかなり高度なメ

ッセージ拡張が行われ、したがって、メッセージ拡張を管理可能な2対1に低減するNTRUの2段階バージョンに関する以下の第4.3節を参照されたい。

まず、実験による証拠によって得られた3つの値と、スプリアス鍵アタックを妨げるのに十分なラッピングが確保されるように選択された第4の値から始める。

$$c_1=0.08, \quad c_2=0.24, \quad W=10 \quad \kappa = 1.5^{1/100} = 1.0040628823$$

c_1 および c_2 の値は所望の範囲の広範囲な数値試験によって決定されているが、これらの値を確率的に正当化するにはどうすべきかについてかなり良好な考え方がある。上記の第4.1節でラッピング係数 $W=10$ について論じた。最後に、第3.3節の注で格子整約定数 κ の選択について論じた。ただし、格子整約技法の将来の改良に備えるために、セキュリティを意識するユーザはこの代わりに $\kappa = 1.3^{1/100}$ を選択し、他のパラメータをわずかに変更することができる。

まず選択肢 $p=2$ について考える。第3.3.5節の不等式(10)によって、次式を選択する必要があることがわかる。

$$(K+1)N \geq 1009.79$$

したがって、以下の値を選択する。

$$N=167 \text{ および } K=6$$

(N と $(N-1)/2$ が共に素数であると好都合である。ただし、これは必要なことではない)。この選択によって、残りの係数を選択するための十分な許容差が与えられる。

L_ϕ を第2.2節と同様に選択し、 $d=20$ とし、したがって、
 $\# L_\phi = 167! / 20! \cdot 20! \cdot 127! \approx 2^{165.85}$ となる。

これはmeet-in-the-middleアタックに対する十分なセキュリティを与える。さらに、 $L_\phi = \sqrt{2d} - 6 = 325$ とし、

これらの選択肢を(11)に代入すると、 $L_g > 414.07$ が与えられる。い

くらかの許容差を与えるために $\gamma = 167$ を選択する。これによって、 L_g の期待値は 622.98 に等しくなる。最後に、第3.3.5節の5つの基本不等式により、 q が次式を満たさなければならないことがわかる。

$$2^{13.6924} < q \leq \max \{2^{14.2766}, 2^{14.7278}, 2^{14.6238}, 2^{52.481}\}$$

(もちろん、第3.3.5節の不等式 (6_k) は実際には、各 $1 \leq k \leq 6$ ごとに1つの6つの不等式である。) したがって、 $q = 2^{14} - 1 = 16383$ を選択することができる。(gcd(p, q) = 1が必要であることに留意されたい。) 簡単に言えば、第3.3節の仮説を仮定した場合、以下のパラメータによって安全なNTRUPKCSが与えられる。

$$N=167, k=6, q=16383=2^{14}-1, p=2, r=167, d=20, s=3$$

この場合、第2.2節で説明したように、 L_ϕ 、 L_g 、 L_m が選択

される。これらのパラメータでは以下の値が得られる。

$$\text{公開鍵長} = N k \log_2 q = 14028 \text{ ビット}$$

$$\text{専用鍵長} = N \log_2 p r = 1400 \text{ ビット}$$

$$\text{メッセージ拡張} = \log q / \log p = 14 \text{ 対 } 1$$

同様な分析を使用して、より大きな値の p を有する第2の1組の安全なNTRUパラメータを構築する。すべての演算が 2^{16} よりも小さな数に対して行われ、 q が2のべき乗であり、したがって、剰余を含む q による除算が簡単なシフト演算になるので、これらのパラメータは既存のマイクロプロセッサにうまく適合するように思われる。以下の値を選択する。

$$N=167, K=6, q=2^{16}, p=3, r=354, d=40, s=7$$

これらのパラメータは $\#L_\phi = 167! / 40! \cdot 40! \cdot 87!$

$\approx 2^{239.3}$ と、以下の値を与える。

$$\text{公開鍵長} = N K \log_2 q = 16032 \text{ ビット}$$

$$\text{専用鍵長} = N \log_2 p r = 1678 \text{ ビット}$$

$$\text{メッセージ拡張} = \log q / \log p = 10.1 \text{ 対 } 1$$

§ 4.3 2段階NTRUおよび改良型メッセージ拡張

第4.2節に示したサンプル・パラメータのNTRUPKCSはかなり大き

なメッセージ拡張を有する。この拡張を減少する1つの方法は、より大きな値の p を使用することであるが、この場合、 $(K+1)N$ の値が著しく大きくなり、そのため両方の鍵サイズが大きくなり計算効率が低下する。

メッセージ拡張を減少する他の方法は、実際のメッセージを符号化するための1種のワнтаイム・パッドとして各NTRUメッセージを使用することである。NTRUのこの2段階バージョンでは、符号化側のC a t h yがランダム多項式 $m \in L_m$ を選択し、これに

対して、C a t h y の実際の平文メッセージ M としては、 q を法とする任意の多項式が許容される。C a t h y は、メッセージを符号化するために、以下の2つの等式を計算する。

$$e = \sum_{i=1}^K p_i \otimes h_i + m \pmod{q} \text{ および } E \equiv m \otimes h_1 + M \pmod{q}$$

符号化メッセージは対 (e, E) である。

復号プロセスは前述のプロセスと類似しているが、1つの余分のステップを含む。したがって、復号側のD a nは、第1.4節で説明した手順に従って多項式 m を計算する。D a nは次いで、次式を計算することによってメッセージを再生する。

$$E - m \otimes h_1 \pmod{q}$$

平文メッセージ M の長さが $N \log_2 q$ ビットであり、それに対して符号化メッセージ (e, E) の長さが $2N \log_2 q$ ビットであり、したがってメッセージ拡張は2対1に減少すると考えられる。

別の点について述べる。C a t h y は同じ多項式および法を使用して m と M の両方を符号化している。これによってセキュリティが損なわれることは考えられないが、セキュリティを強化した場合、

C a t h y は異なる (公開) 多項式 H および法 Q について $E \equiv m \otimes H + M \pmod{Q}$ を計算することができる。

§ 4.4 理論上の動作仕様

この節では、NTRU PKCSの理論上の動作特性について考える。4つの

整数パラメータ (N 、 K 、 p 、 q) と、第2. 2節で説明したように、それぞれ、整数 r 、 d 、 s によって決定される、3つの集合 L_g 、 L_ϕ 、 L_m と、実験的に決定される定数 c_1 、 c_2 、 k と、ラッピング定数 W がある。セキュリティを保証するには、こ

れらのパラメータとして、第3. 3. 5節にリストした不等式を満たすパラメータを選択しなければならない。以下の表は、これらのパラメータで表したNTRU PKCS動作特性を要約したものである。

平文ブロック	$N \log_2 p$ ビット
符号化テキスト・ブロック	$N \log_2 q$ ビット
符号化速度*	$O(KN^2)$ 回の演算
復号速度	$O(N^2)$ 回の演算
メッセージ拡張	$\log_p q$ 対 1
専用鍵長	$N \log_2 p r$ ビット
公開鍵長	$KN \log_2 q$ ビット

*厳密には、 $4KN^2$ 回の加算と、 KN 回の、剰余を含む q による除算

第4. 4節で説明した2段階NTRUについては以下の項目が異なる。

平文ブロック	$N \log_2 q$ ビット
符号化テキスト・ブロック	$2N \log_2 q$ ビット
メッセージ拡張	2 対 1

§ 4. 5 実施上の他の考慮すべき点

NTRUを実施する際に考慮すべき他のいくつかの因子について簡単に述べる。

(1) $\gcd(q, p) = 1$ であることが重要である。基本的に N

TRUはこの要件がなくても働くが、実際には、 $\gcd(q, p) > 1$ である場合、セキュリティが低下する。極端な範囲では、 $p \mid q$ である場合、(exer

c i s e) 符号化メッセージ e は $e \equiv m \pmod{p}$ を満たし、NTRUのセキュリティは完全に失われる。

(2) 大部分の f が p および q を法とする逆数を有することが望ましい。というのは、そうでない場合、鍵の作成が困難になるからである。第1の要件は $\gcd(f(1), pq) = 1$ であるが、これが、ある選択された f に対して無効であった場合、符号作成者はこの代わりにたとえば、 $f(X) + 1$ または $f(X) - 1$ を使用することができる。 $\gcd(f(1), pq) = 1$ と仮定すると、 N として素数を選択し、 p および q を除する各素数 P について、 $(\mathbb{Z}/N\mathbb{Z})^*$ 中の P のオーダーを大きくし、たとえば $N-1$ または $(N-1)/2$ にした場合、ほぼすべての f 、が必要な逆数を有する。たとえば、これは、 $(N-1)/2$ 自体が素数である（すなわち、 N が Sophie Germain 素数である）場合は確実に真である。このような素数の例には 107 および 167 が含まれる。

§ 5 NTRUの適度なセキュリティ・パラメータ

現実には、高速および／または低メモリ要件が重要であり、適度なセキュリティ・レベルが受け入れられる多数の状況がある。この場合、実際の格子整約方法 [11, 12] はCPUを酷使し、そればかりでなく、次元200ないし300の格子に対して格子整約を実行するのに長いコンピュータ時間を必要とする。もちろん、この場合の「長い」は相対的な語であるが、300次元格子整約を実行して1セントの数分の1に相当するコストを削減しても恐らく無効であり、現行の方法を使用してこのような格子整約を短時間（たと

えば2、3分間）で実行すると（完全に実現不能ではない場合）コストが非常に高くなることは確実である。したがって、大次元格子アタックを可能にする必要のある状況で使用できる1組のNTRUパラメータを作成すると有効である。

格子アタックによってもたらされるパラメータ制約をなくした場合、残るのは以下の復号制約と、

$$c \geq p L_g L_\phi (K+1) < q \quad (4)$$

f 、 g 、 ϕ の探索空間が暴力的（多分 meet-in-the-middle）アタックを防止するほど大きいという条件だけである。話を簡単にするために、

$K=1$ を選択する。 f 、 g 、 ϕ がすべて、集合 L_ϕ 、すなわち、 d 個の係数が1に等しく、 d 個の係数が -1 に等しく、他の $N-2d$ 個の係数が0に等しい多項式の集合に含まれるものとみなす。(厳密には、 f は p および q を法として可逆である必要があるので、 f を余分の1つの係数を有するものとみなすが、これは後続く分析にほとんど影響を与えず、したがってこれを無視することにする。)

$c_2=0$ 。24を通常どおりに使用すると、復号制約は単に次式のようなになる。

$$q > 2pd \quad (4)$$

他の制約を次式に示す。

$$\binom{N}{d; d; N-2d} = \frac{N!}{(d!)^2(N-2d)!} \approx 2^{2\sigma}$$

上式で、 σ は必要なセキュリティ・レベルである。適度なセキュリティの処理系の場合、セキュリティ・レベルは約 2^{40} でほぼ十分

であり、したがって $\sigma \approx 40$ を選択する。

以下の表は、NTRUの適度なセキュリティの処理系の受け入れられる動作パラメータを示す。セキュリティを評価する際、利用可

能な格子アタックが次元 $2N$ の格子を使用することに留意されたい。また、 q のリストされた値は最小許容値であるが、 $\gcd(p, q)$ を満たす、これよりもいくらか大きな q も受け入れられることに留意されたい。特に、 $q=64$ を選択することによってとりわけ高速な処理系を使用することができる。

N	d	σ	p	q
107	9	41.11	2	37
107	9	41.11	3	55
167	7	38.98	2	29
167	7	38.98	3	43
263	7	43.72	2	29
263	7	43.72	3	43

最後に、鍵サイズが非常に小さくなると考えられる。

公開鍵: $N \log_2(q)$ ビット

専用鍵: $2N \log_2(p)$ ビット

たとえば、 $(N, d, p, q) = (167, 7, 3, 64)$ は、それぞれ長さ1002ビットおよび530ビットの公開鍵および専用鍵を含むシステムを形成する。

§ 6 他のPKCSとの比較

現在、因数分解の難点に基づくRivest、Shamir、Adelman (RSA [10]) のシステム、誤り補正符号に基づくMcEliece [9] のシステム、ほぼV直交化された短い基を格子内で見つけることが困難であることに基づくGoldreich、Goldwasser、Halevi (GGH [5]) の最近のシステムを含め、いくつかの公開鍵暗号システムが文献に記載されている。

NTRUシステムは、環R内の星印乗算を(特殊な種類の)行列の乗算として公式化することができ、次いで、システムでの符号化を行列乗算 $E = AX + Y$ (A は公開鍵である)として書くことができるという点で、McElieceのシステムと共通するいくつかの特徴を有する。2つのシステムの間の小さな違いは、NTRU符号化では、 Y がメッセージであり、 X がランダム・ベクトルであるが、McElieceシステムではこれらの割当てが逆になることである。しかし、実際の違いは、復号を可能にする基本トラップドアである。McElieceシステムの場合、行列 A が誤り補正(ゴッパ)符号に関連付けられ、ランダム寄与がゴッパ符号によって「補正される」ほど小さいために復号が作用する。NTRUの場合、行列 A は巡回行列であり、復号は、 A の、特殊な形式を有する2つの行列の積へ分解と、 $\text{mod } q$ から $\text{mod } p$ へのリフティングに依存する。

我々の知るかぎりでは、NTRUシステムにはRSAシステムとの共通点がほとんどない。同様に、NTRUシステムは格子整約アタックを防止するようにセットアップしなければならないが、その基本復号方法は、復号が短い格子基の知識に基づいて行われるGGHシステムとはかなり異なるものである。なお、GG

Hは実際には、McElieceシステムに類似している。これは、どちらの場合でも、小さなランダム寄与を認識しなくすことによって復号が実行されるからである。これに対して、NTRUは、可分性（すなわち、合同）を考慮することによってずっと大きなランダム寄与をなくす。

以下の表は、RSA暗号システム、McEliece暗号システム、GGH暗号システム、NTRU暗号システムのいくつかの理論的動作特性を比較したものである。それぞれの場合において、数Nは固有セキュリティ/メッセージ長パラメータを表す。

	NTRU	RSA	McEliece	GGH
符号化速度	N^2	N^2	N^2	N^2
復号速度	N^2	N^3	N^2	N^2
公開鍵	N	N	N^2	N^2
専用鍵	N	N	N^2	N^2
メッセージ拡張	2-1	1-1	2-1	1-1

付録A. 基本補題

以下の結果は格子アタックを最適化するうえで有用である。補題A. 1. $\alpha + \beta = 1$ を有するすべてのA、B、 α 、 β について、

$$\inf_{x>0} Ax^\alpha + Bx^{-\beta} = \frac{A^\beta B^\alpha}{\alpha^\alpha \beta^\beta}$$

が成立し、 $x = \beta B / \alpha A$ で下限が生じる。

証明 $f(x) = Ax^\alpha + Bx^{-\beta}$ とする。この場合、 $f'(x) = \alpha Ax^{\alpha-1} - \beta Bx^{-\beta-1} = x^{\beta+1}(\alpha Ax - \beta B)$ である。したがって、絶対最小値は $x = \beta B / \alpha A$ で生じる。（ $x \rightarrow 0^+$ および $x \rightarrow \infty$ のときに $f(x) \rightarrow \infty$ であることに留意されたい。）

参考文献

1. M. Blum, S. Goldwasser 著「An efficient probabilistic public-key encryption scheme which hides all partial information」Advances in Cryptology: Proceedings of CRYPTO 84, Lecture Notes in Computer Science, 第 196 巻、Springer-Verlag, 1985 年、289 ページ～299 ページ
2. H. Cohen 著「A course in computational algebraic number theory」Graduate Texts in Math, 第 138 巻、Springer Verlag, ベルリン、1993 年
3. D. Coppersmith, A. Shamir 著「Lattice attacks on NTRU」前刷り、1997 年 4 月 5 日、Eurocrypt 97 にて発表
4. W. Diffie, M. E. Hellman 著「New directions in cryptography」IEEE Trans. on Information Theory 22 (1976 年)、644 ページないし 654 ページ
5. O. Goldreich, S. Goldwasser, S. Halevi 著「Public-key

cryptosystems from lattice reduction problems」 MIT-Laboratory for Computer Science 前刷り、1996 年 11 月

6 . S. Goldwasser および A. Micali 著 「Probabilistic encryption」 J. Computer and Systems Science 28 (1984 年)、270 ページないし 299 ページ

7 . J. Hoffstein、J. Pipher、J.H. Silverman 著 「NTRU: A new high speed public key cryptosystem、前刷り、Crypto 96 のランプ・セッションにて発表

8 . A.K. Lenstra、H.W. Lenstra、L. Lovsz 著 「Factoring polynomials with polynomial coefficients」 Math. Annalen 261 (1982 年)、515 ページないし 534 ページ

9 . R.J. McEliece 著 「A public-key cryptosystem based on algebraic coding theory」、JPL Pasadena, DSN Progress Reports 第 42 巻ないし第 44 巻 (1978 年)、114 ページないし 116 ページ

10 . R.L. Rivest、A. Shamir、L. Adleman 著 「A method for obtaining digital signatures and public key cryptosystems」 Communications of the ACM 21 (1978 年)、120 ページないし 126 ページ

11 . C.P. Schnorr 著 「Block reduced lattice bases and successive minima」 Combinatorics, Probability and Computing 3 (1994 年)、507 ページないし 522 ページ

12 . C.P. Schnorr、H.H. Hoerner 著 「Attacking the Chor Rivest cryptosystem by improved lattice reduction」 Proc. EUROCRYPT 1995, Lecture Notes in Computer Science 921, Springer-Verlag、1995 年、1 ページないし 12 ページ

13 . D. Stinson 著 「Cryptography: Theory and Practice」 CRC Press、

Boca Raton、1995 年

【図1】

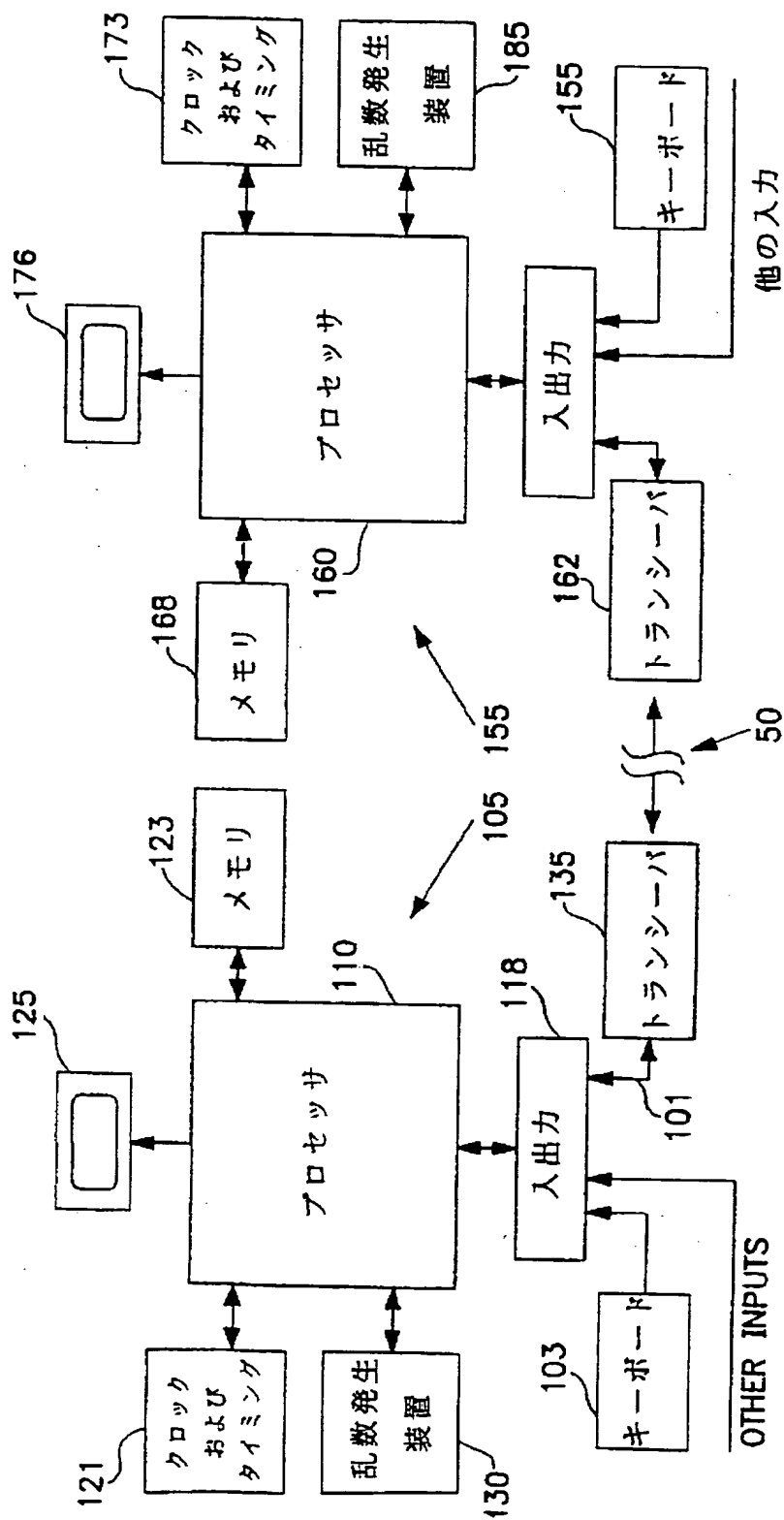


FIG. 1

【図2】

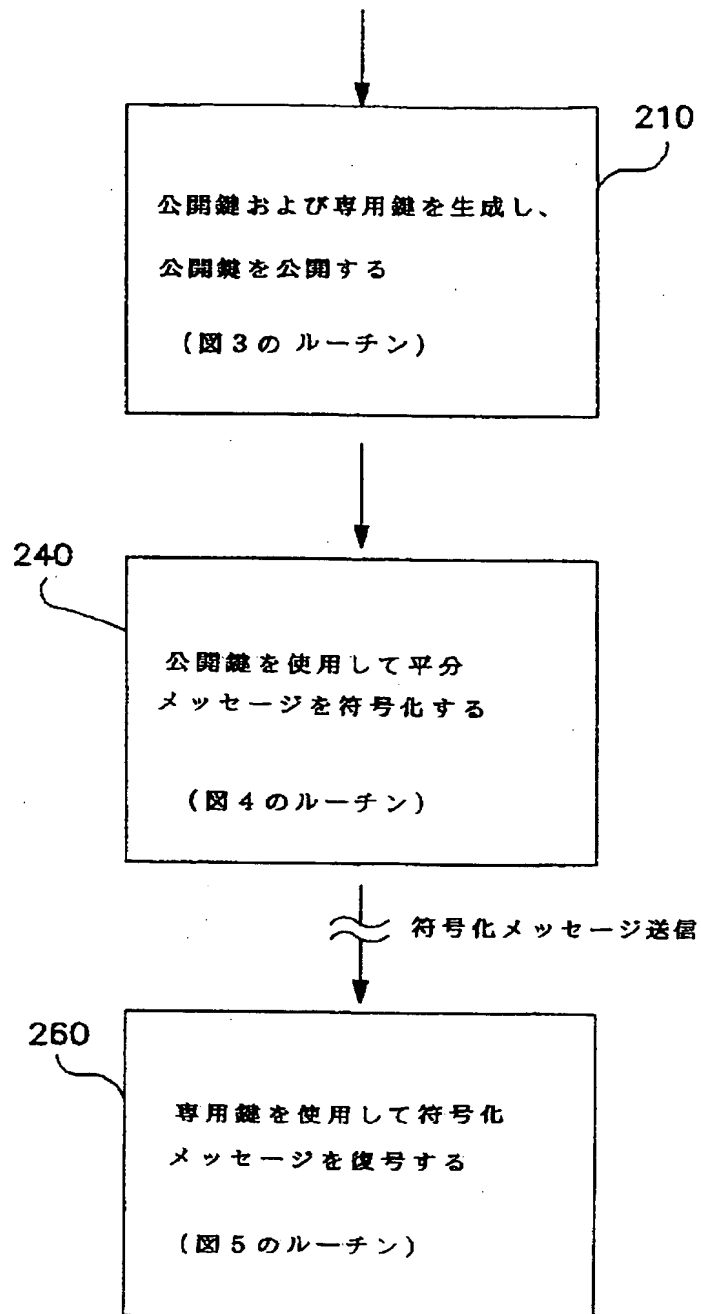


FIG. 2

【図3】

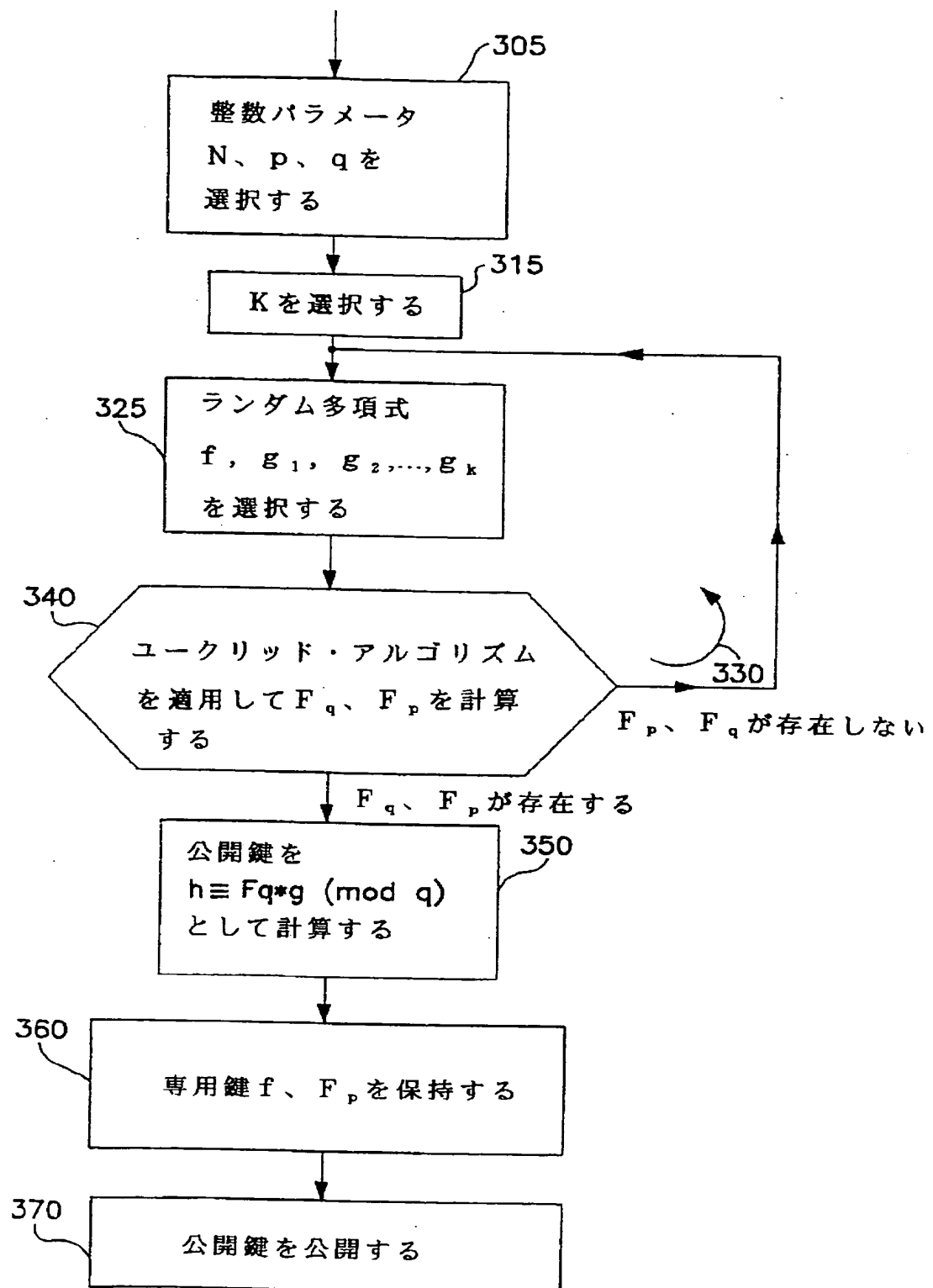


FIG. 3

【図4】

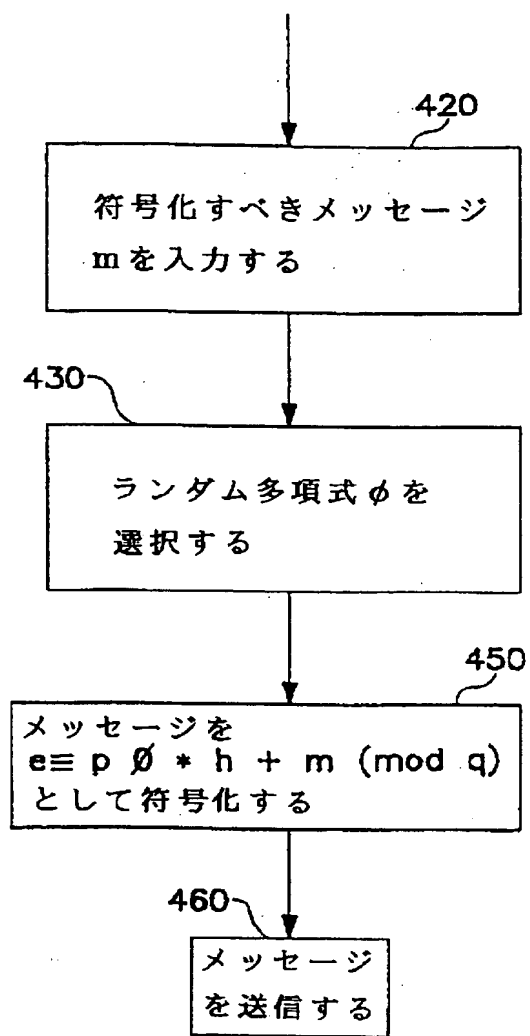


FIG. 4

【図5】

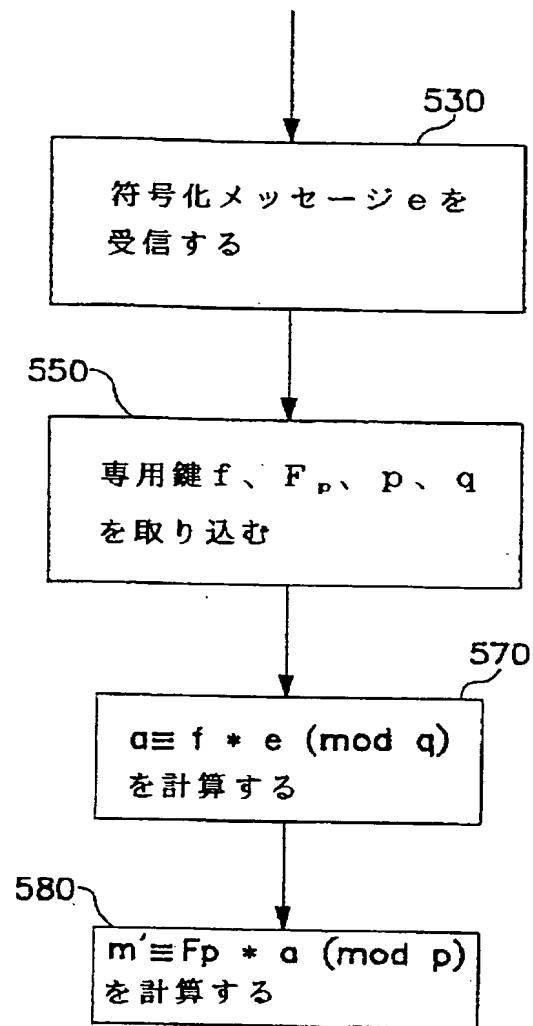


FIG. 5

【図6】

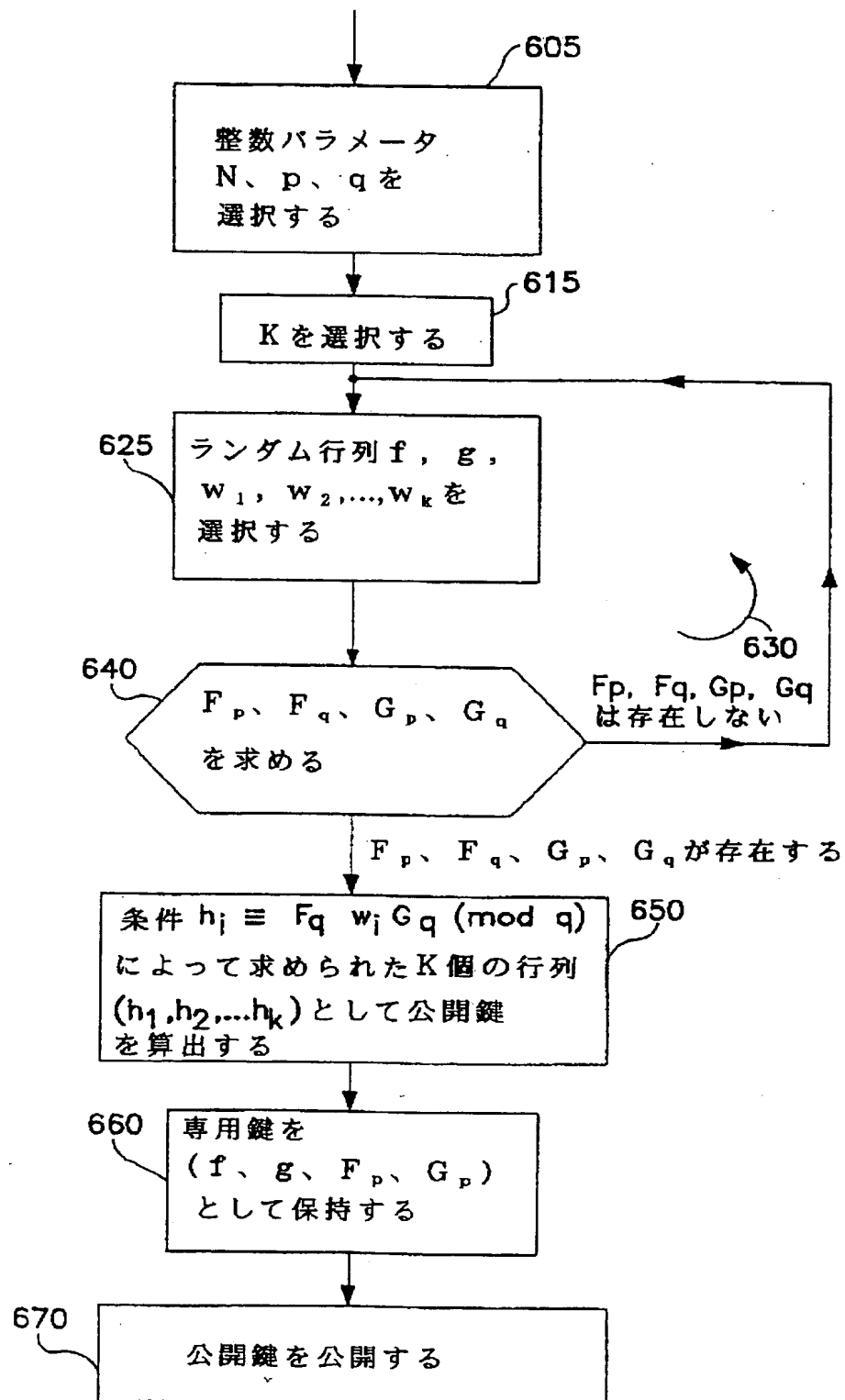


FIG. 6

【図7】

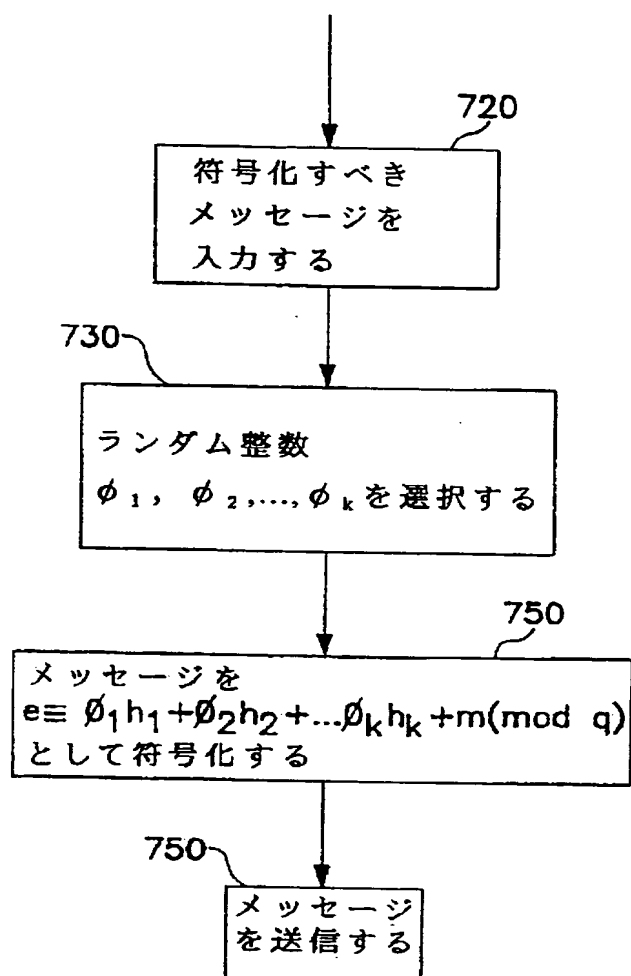


FIG. 7

【図8】

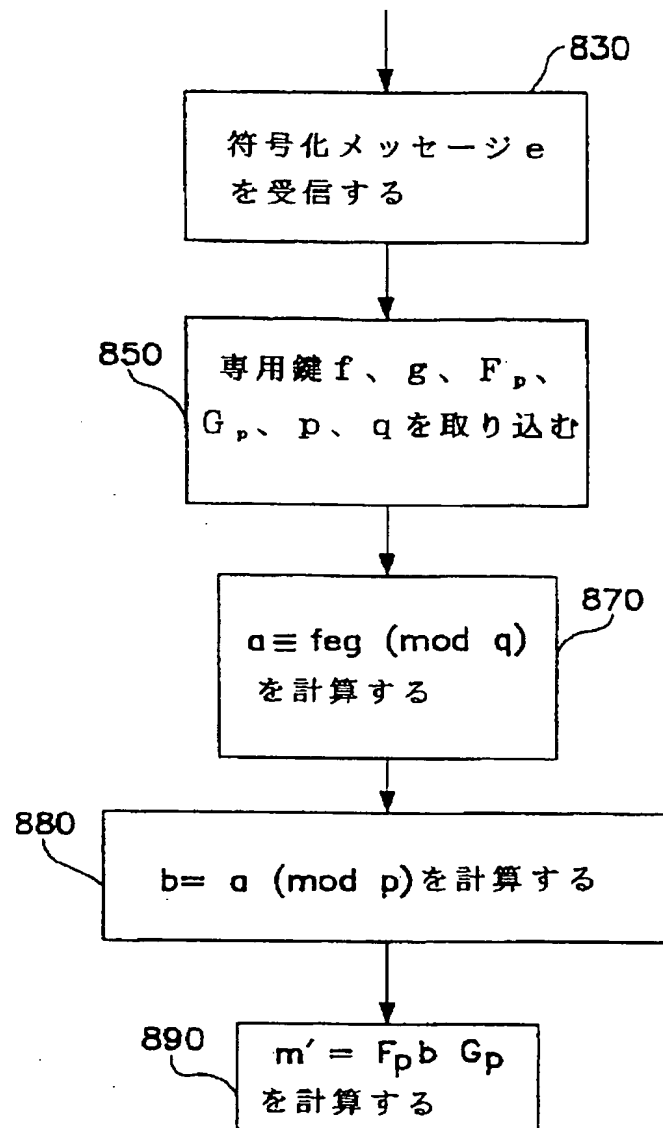


FIG. 8

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US97/15826

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) : H04K 1/00

US CL : 380/30, 28, 49

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 380/30, 28, 49

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A, P	US 5,600,725 A (RUEPPEL et al.) 04 February 1997.	1-56
A, P	US 5,577,124 A (ANSHEL et al.) 19 November 1996.	1-56
A	US 5,375,170 A (SHAMIR) 20 December 1994.	1-56
A	US 5,351,297 A (MIYAJI et al.) 27 September 1994.	1-56

☐ Further documents are listed in the continuation of Box C.☐ See patent family annex.

Special categories of cited documents:	
A document defining the general state of the art which is not considered to be of particular relevance	*T* later documents published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
B earlier document published on or after the international filing date	*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
L document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
O document referring to an oral disclosure, use, exhibition or other means	*A* document member of the same patent family
P document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

05 DECEMBER 1997

Date of mailing of the international search report

27 JAN 1998

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231

Facsimile No. (703) 305-3230

Authorized officer

DAVID CAIN

Telephone No. (703) 305-1136

フロントページの続き

(72)発明者 バイファー ジル
アメリカ合衆国, ロード アイランド
02860, ポータケット, レスター ウェイ,
3番地

(72)発明者 シルヴァーマン ジョセフ エイチ,
アメリカ合衆国, マサチューセッツ
02192, ニーダム, ノース ヒル アヴェ
ニュー, 57番地

THIS PAGE BLANK (USPTO)

THIS PAGE BLANK (USPTO)